



Correctness of Linear Logic Proof Structures is NL-Complete

Paulin Jacobé de Naurois, Virgile Mogbil

► To cite this version:

Paulin Jacobé de Naurois, Virgile Mogbil. Correctness of Linear Logic Proof Structures is NL-Complete. Theoretical Computer Science, 2011, 412 (20), pp. 1941-1957. 10.1016/j.tcs.2010.12.020 . hal-00360894

HAL Id: hal-00360894

<https://hal.science/hal-00360894>

Submitted on 12 Feb 2009

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Correctness of Linear Logic Proof Structures is *NL*-Complete

(rapport interne LIPN - Decembre 2008)

Paulin Jacobé de Naurois* and Virgile Mogbil*
LIPN – UMR7030, CNRS – Université Paris 13,
99 av. J-B Clément, F-93430 Villetaneuse, France
denaurois@lipn.univ-paris13.fr
virgile.mogbil@lipn.univ-paris13.fr

Abstract

We provide new correctness criteria for all fragments (multiplicative, exponential, additive) of linear logic. We use these criteria for proving that deciding the correctness of a linear logic proof structure is *NL*-complete.

Introduction

The *proof nets* [Gir87, DR89] of Linear logic (LL) are a parallel syntax for logical proofs without all the bureaucracy of sequent calculus. They are a non-sequential graph-theoretic representation of proofs, where the order in which some rules are used in a sequent calculus derivation, when irrelevant, is neglected. The unit-free multiplicative proof nets are inductively defined from sequent calculus rules of unit-free Multiplicative Linear Logic (MLL). The MLL *proof structures* are freely built on the same syntax as proof nets, without any reference to a sequent calculus derivation. The same holds for MELL and MALL proof nets and proof structures with respect to MELL and MALL sequent calculus.

In LL we are mainly interested in the following decision problems: Deciding the provability of a given formula, which gives the expressiveness of the logic; deciding if two given proofs reduce to the same normal form, i.e. the cut-elimination problem which corresponds to program equivalence using the Curry-Howard isomorphism; and deciding the correctness of a given proof structure, i.e. whether it comes from a sequent calculus derivation. For this last decision problem, one uses a *correctness criterion* to distinguish proof nets among proof structures. We recall the following main results [Kan92, Mai06] and we complete (in bold) the correctness cases:

fragment		decision problem		
	units	provability	cut-elimination	correctness
MLL	no	<i>NP</i> -complete	<i>P</i> -complete	NL-complete
MELL	yes	open	non-elementary	
MALL	no	<i>PSPACE</i> -complete	<i>coNP</i> -complete	

Correctness is equivalent to provability for unit only MLL because proof nets are formulae syntactic trees. However it is not so obvious for the propositional case as one can observe following the long story of correctness criteria:

- Long-trip [Gir87] is based on travels and was the first one.
- Acyclic-Connected [DR89] is a condition is based on switchings i.e. the choice of one premise for each \otimes connective. The condition is that all the associated graphs are trees. A naive implementation of Acyclic-Connected uses exponential time.

*Work supported by project NOCoST (ANR, JC05.43380)

- Contractibility [Dan90] is done in quadratic time by repeating two graph rewriting rules until one obtains a simple node.
- Graph Parsing [Laf95] is a strategy for Contractibility which is implemented in linear time as a sort of unification [Gue99].
- Dominator Tree [MO00, MO06] is a linear time correctness criterion for essential nets, to which proof structures correctness reduces in linear time.
- Ribbon [Mel04] is a topological condition requiring homeomorphism to the disk.

For other fragments of Linear Logic, some of these criteria apply or are extended as for MELL¹ [Dan90, GM01] or MALL [Gir96, dF03, HvG05].

A feature of these criteria is that they successively lower the complexity of sequential, deterministic algorithms deciding correctness for MLL. Switching from proof structures to paired graphs, that is undirected graphs with a distinguished set of edges, we give a new correctness criterion for MLL and more generally for MELL. This new correctness criterion gives, for the first time, a lower bound for the correctness decision problem for both MLL (MLL-CORR) and MELL (MELL-CORR). This lower bound yields an exact characterization of the complexity of this problem, and induces naturally efficient parallel and randomized algorithms for it. The classical inclusion $NL \subseteq P$ induces a deterministic polynomial time version of our algorithm; note however that there is little hope for this to be linear time.

Our new criterion also induces an NL algorithm for the correctness problem for MALL proof structures (MALL-CORR) as defined in [HvG05], thus establishing as well the NL -hardness of this problem.

The paper is organized as follows: we recall preliminary definitions and results in linear logic and complexity theory in Section 1. Section 2 is devoted to the exposition of a new correctness criterion for unit-free MLL and MELL with units (Theorem 2.6). Proposition 2.7 establishes the NL -hardness of MLL-CORR and Proposition 2.9 the NL -membership of MELL-CORR. Section 3 is devoted to the proof of the NL -membership of MALL-CORR. The NL -completeness of MLL-CORR, MELL-CORR and MALL-CORR is established in Theorem 3.20.

1 Background

1.1 MLL and Proof Structures

Roman capitals A, B stand for MLL formulae, which are given by the following grammar, where \otimes and \wp are duals for the negation $^\perp$, accordingly to De Morgan laws:

$$\text{MLL:} \quad F ::= A \mid A^\perp \mid F \otimes F \mid F \wp F$$

Greek capitals Γ, Δ stand for sequents, which are multiset of formulae, so that exchange is implicit. The MLL sequent calculus is given by the following rules:

$$\frac{}{\vdash A, A^\perp} (ax) \quad \frac{\vdash \Gamma, C \quad \vdash \Delta, C^\perp}{\vdash \Gamma, \Delta} (cut) \quad \frac{\vdash \Gamma, A \quad \vdash \Delta, B}{\vdash \Gamma, \Delta, A \otimes B} \otimes \quad \frac{\vdash \Gamma, A, B}{\vdash \Gamma, A \wp B} \wp$$

Definition 1.1. A MLL *skeleton* is a directed acyclic graph (DAG) whose edges are labelled with MLL formulae, and whose nodes are labelled, and defined with an arity and co-arity as follows:

node label		atom		cut		\otimes		\wp	
arity	edges	0	\emptyset	2	A, A^\perp	2	A, B	2	A, B
coarity	edges	1	A	0	\emptyset	1	$A \otimes B$	1	$A \wp B$

¹As usual M, A and E denote respectively for Multiplicative, Additive and Exponential fragments of LL

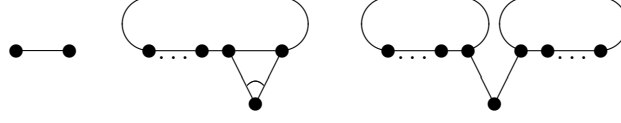


Figure 1: Paired graph constructors associated to MLL proof nets: ax -link, \wp -link and \otimes (cut)-link.

We allow edges with a source but no target (i.e pending or dandling edges), they are called the *conclusions* of the proof-structure. The set of conclusions of a MLL skeleton is clearly a MLL sequent. We also denote as *premises* of a node the edges incident to it, and *conclusion* of a node its outgoing edge.

For a given node x of arity 2, its left (respectively right) parent is denoted x^l (resp. x^r).

An *axiom-link*, or simply *link* on a MLL skeleton \mathcal{S} is a bidirected edge between complementary atoms in \mathcal{S} , i.e. atoms labeled with dual literals P and P^\perp .

A *linking* on \mathcal{S} is a partitioning of the atom nodes of \mathcal{S} into links, i.e. a set of disjoint links whose union contains every atom of \mathcal{S} .

A MLL *proof structure* is (\mathcal{S}, λ) , where \mathcal{S} is a MLL skeleton and λ a linking on \mathcal{S} .

A MLL *proof net* is a MLL proof structure inductively defined as follows:

- (ax) : (\mathcal{S}, λ) where $\mathcal{S} = (\{A, A^\perp\}, \emptyset)$, $\lambda = \{(A, A^\perp)\}$ is a MLL proof net with conclusions A, A^\perp .
- \wp : if (\mathcal{S}, λ) is a MLL proof net with conclusions Γ, A, B , then (\mathcal{S}', λ) , where \mathcal{S}' is \mathcal{S} extended with a \wp -node of premises A and B is a MLL proof-net with conclusions $\Gamma, A \wp B$.
- \otimes : if $(\mathcal{S}_1, \lambda_1)$ with conclusions Γ, A and $(\mathcal{S}_2, \lambda_2)$ with conclusions Δ, B are disjoint MLL proof nets, $(\mathcal{S}, \lambda_1 \uplus \lambda_2)$ where \mathcal{S} is $\mathcal{S}_1 \uplus \mathcal{S}_2$ extended with a \otimes -link of premises A and B is a MLL proof net with conclusions $\Gamma, A \otimes B, \Delta$.
- (cut) : if $(\mathcal{S}_1, \lambda_1)$ with conclusions Γ, A and $(\mathcal{S}_2, \lambda_2)$ with conclusions Δ, A^\perp are disjoint MLL proof nets, $(\mathcal{S}, \lambda_1 \uplus \lambda_2)$ where \mathcal{S} is $\mathcal{S}_1 \uplus \mathcal{S}_2$ extended with a cut-link of premises A and A^\perp is a MLL proof net with conclusions Γ, Δ .

The inductive definition of MLL proof nets corresponds to a graph theoretic abstraction of the derivation rules of MLL; any proof net is sequentializable, i.e. corresponds to a MLL derivation: given a proof net P of conclusion Γ , there exists a sequent calculus proof of $\vdash \Gamma$ which infers P .

Definition 1.2. A *paired graph* is an undirected graph $G = (V, E)$ with a set of *pairs* $C(G) \subseteq E \times E$ which are pairwise disjoint couples of edges with the same target, called a *pair-node*, and two (possibly distinct) sources called the *premise-nodes*.

A *switching* S of G is the choice of an edge for every pair of $C(G)$. With each switching S is associated a subgraph $S(G)$ of G : for every pair of $C(G)$, erase the edges which are not selected by S . When S selects the (abusively speaking) left edge of each pair, $S(G)$ is denoted as $G[\forall \mapsto \cdot]$. Also, $G[\forall \mapsto \cdot]$ stands for $G \setminus \{e, e' \mid (e, e') \in C(G)\}$.

Remark 1.3. Without loss of generality we allow tuples of edges, i.e. $C(G) \subseteq \bigcup_{n \in \mathbb{N}} E$. A tuple of edges incident to a node x can be seen as a binary tree rooted at x with all ingoing edges being coupled.

Let $\mathcal{S} = (V, E)$ be a MLL skeleton. To \mathcal{S} , we associate the paired graph $G_{\mathcal{S}} = (V, E)$, where $C(G_{\mathcal{S}})$ contains the premises of each \wp -link of \mathcal{S} .

To a MLL proof structure (\mathcal{S}, λ) , we associate the paired graph $G_{(\mathcal{S}, \lambda)} = G_{\mathcal{S}} \uplus \lambda$, where $C(G_{(\mathcal{S}, \lambda)}) = C(G_{\mathcal{S}})$ (Figure 1).

For a pair of edges $(v, x), (w, x)$, we adopt the representation of Figure 1, where the two edges of the pair are joined by an arc. We define the following graph rewriting rules \rightarrow_c of Figure 2 on

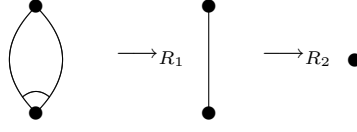


Figure 2: Contraction rules \rightarrow_c

paired graphs where all the nodes are distinct and rule \rightarrow_{R_2} applies only for a non-pair edge: We denote by $G \rightarrow_c^* \bullet$ the fact that G contracts to a single vertex with no edge.

Definition 1.4. A MLL proof structure (\mathcal{S}, λ) is *DR-correct* if for all switching S of $G_{(\mathcal{S}, \lambda)}$, the graph $S(G_{(\mathcal{S}, \lambda)})$ is acyclic and connected.

A MLL proof structure (\mathcal{S}, λ) is *contractile* if $G_{(\mathcal{S}, \lambda)} \rightarrow_c^* \bullet$.

Theorem 1.5. [DR89, Dan90] A MLL proof structure (\mathcal{S}, λ) is a MLL proof net iff (\mathcal{S}, λ) is DR-correct iff (\mathcal{S}, λ) is contractile \square

We define the following decision problem MLL-CORR:

GIVEN: A MLL proof structure (\mathcal{S}, λ)

PROBLEM: Is (\mathcal{S}, λ) a MLL proof net?

1.2 MELL and Proof Structures

The definition of MELL formulae follows that of MLL formulae in Section 1.1, with ! and ? duals for the negation $^\perp$, as well as the neutral elements 1 and \perp :

$$\text{MELL:} \quad F ::= A \mid A^\perp \mid F \otimes F \mid F \wp F \mid !F \mid ?F \mid 1 \mid \perp$$

The MELL sequent calculus contains the rules of the MLL sequent calculus, as well as the following rules:

$$\frac{\vdash \Gamma}{\vdash \Gamma, \perp} \perp \quad \frac{}{\vdash 1} 1 \quad \frac{\vdash \Gamma}{\vdash \Gamma, ?A} ?W \quad \frac{\vdash \Gamma, ?A, ?A}{\vdash \Gamma, ?A} ?C \quad \frac{\vdash \Gamma, A}{\vdash \Gamma, ?A} ?D \quad \frac{\vdash ?\Gamma, A}{\vdash ?\Gamma, !A} !P$$

Definition 1.6. MELL *skeletons* are defined similarly to MLL skeletons (Definition 1.1), with the following additional nodes, where the ?W-node subsumes both ?W and \perp MELL sequent calculus rules:

node label		1		?W		?C		?D		!P	
arity	edges	0	\emptyset	0	\emptyset	2	$?A, ?A$	1	A	1	A
coarity	edges	1	1	1	$\perp, ?A$	1	$?A$	1	$?A$	1	$!A$

Definition 1.7. An *exponential box* is a MELL skeleton whose conclusions are all ?-formulae but one, its *principal door*, which is conclusion of a !P-node. Similarly, a *weakening box* is a MELL skeleton with a distinguished conclusion, its *principal door*, which is conclusion of a ?W-node. A *box* is either an exponential or a weakening box.

Definition 1.8. A MELL *boxed structure* (\mathcal{S}, B) is given by a MELL skeleton \mathcal{S} and a set of exponential and weakening boxes $B = \{B_1, \dots, B_k\}$. Moreover, boxes may nest but may not partially overlap. For a given node in \mathcal{S} , its associated box (if there is any) is the smallest box in B that contains it. A unique exponential (respectively weakening) box is associated to each !P-(resp. ?W)-node. The set B of boxes is identified with a box mapping B which, for a given node in \mathcal{S} , returns its associated box if there is any, and \mathcal{S} otherwise.

Definition 1.9. A MELL *proof structure* is $(\mathcal{S}, B, \lambda)$, where (\mathcal{S}, B) is a boxed structure and λ is a linking on \mathcal{S} .

As for MLL, the set of conclusions of a MELL proof structure is by construction a MELL sequent.

Definition 1.10. A MELL *proof net* is a MELL proof structure defined inductively as follows:

- (ax) : $((\{A, A^\perp\}, \emptyset), \emptyset, \{(A, A^\perp)\})$ is a MELL proof net with conclusions A, A^\perp .
- \wp : if $(\mathcal{S}, B, \lambda)$ is a MELL proof net with conclusions Γ, A, C , then $(\mathcal{S}', B, \lambda)$, where \mathcal{S}' is \mathcal{S} extended with a \wp -node of premises A and C , is a MELL proof-net with conclusions $\Gamma, A \wp C$.
- \otimes : if $(\mathcal{S}_1, B_1, \lambda_1)$ with conclusions Γ, A and $(\mathcal{S}_2, B_2, \lambda_2)$ with conclusions Δ, C are disjoint MELL proof nets, $(\mathcal{S}, B_1 \uplus B_2, \lambda_1 \uplus \lambda_2)$, where \mathcal{S} is $\mathcal{S}_1 \uplus \mathcal{S}_2$ extended with a \otimes -link of premises A and C , is a MELL proof net with conclusions $\Gamma, A \otimes C, \Delta$.
- (cut) : if $(\mathcal{S}_1, B_1, \lambda_1)$ with conclusions Γ, A and $(\mathcal{S}_2, B_2, \lambda_2)$ with conclusions Δ, A^\perp are disjoint MELL proof nets, $(\mathcal{S}, B_1 \uplus B_2, \lambda_1 \uplus \lambda_2)$ where, \mathcal{S} is $\mathcal{S}_1 \uplus \mathcal{S}_2$ extended with a cut-link of premises A and A^\perp , is a MELL proof net with conclusions Γ, Δ .
- 1 : $((\{1\}, \emptyset), \emptyset, \emptyset)$ is a MELL proof net with conclusion 1.
- ?W : if $(\mathcal{S}, B, \lambda)$ is a MELL proof net with conclusions Γ , then, for any MELL formula A , $(\mathcal{S}', B \uplus \mathcal{S}', \lambda)$, where \mathcal{S}' is \mathcal{S} extended with a ?W-node with conclusion $?A$ (respectively \perp), is a MELL proof-net with conclusions $\Gamma, ?A$ (respectively Γ, \perp).
- ?C : if $(\mathcal{S}, B, \lambda)$ is a MELL proof net with conclusions $\Gamma, ?A, ?A$, then $(\mathcal{S}', B, \lambda)$, where \mathcal{S}' is \mathcal{S} extended with a ?C-node of premises $?A$ and $?A$, is a MELL proof-net with conclusions $\Gamma, ?A$.
- ?D : if $(\mathcal{S}, B, \lambda)$ is a MELL proof net with conclusions Γ, A , then $(\mathcal{S}', B, \lambda)$, where \mathcal{S}' is \mathcal{S} extended with a ?D-node of premise A , is a MELL proof-net with conclusions $\Gamma, ?A$.
- !P : if $(\mathcal{S}, B, \lambda)$ is a MELL proof net with conclusions $? \Gamma, A$, then $(\mathcal{S}', B \uplus \mathcal{S}', \lambda)$, where \mathcal{S}' is \mathcal{S} extended with a !P-node of premise A , is a MELL proof-net with conclusions $? \Gamma, !A$.

As for MLL, MELL proof nets are induced by MELL sequent calculus proofs.

Definition 1.11. Let $(\mathcal{S}, B, \lambda)$ be a MELL proof structure, with boxes b_1, \dots, b_n . Let $b_0 = \mathcal{S}$. We define as follows the family $\mathcal{G}_{(\mathcal{S}, B, \lambda)} = \{G_{(\mathcal{S}, B, \lambda)}^i\}_{i=0 \dots n}$ of paired-graphs:

- $G_{(\mathcal{S}, B, \lambda)}^i$ contains a node l for every node l of $\mathcal{S} \setminus \{\text{?W-nodes}\}$ with $B(l) = b_i$
- $G_{(\mathcal{S}, B, \lambda)}^i$ contains an edge (l, l') for $l, l' \in \mathcal{S} \setminus \{\text{?W-nodes}\}$ such that $B(l) = B(l') = b_i$ if and only if:
 - there is an edge (l, l') in \mathcal{S} , or
 - there is an edge (l', l) in \mathcal{S} , or
 - $(l, l') \in \lambda$.
- $C(G_{(\mathcal{S}, B, \lambda)}^i)$ contains the premises of each \wp -node and ?C-node l of \mathcal{S} with $B(l) = b_i$.
- Assume b_j is an outermost box included in b_i . A node $\overline{b_j} \in G_{(\mathcal{S}, B, \lambda)}^i$ is associated to b_j , and an edge $(\overline{b_j}, l) \in G_{(\mathcal{S}, B, \lambda)}^i$ for all node l conclusion of a node in b_j and such that $B(l) = b_i$.

Essentially, $G_{(\mathcal{S}, B, \lambda)}^i$ is the paired graph corresponding to the box b_i , where all inner boxes are considered contracted to a single node. Moreover, for the sake of connectivity, the ?W-node (if there is any) corresponding to b_i is removed.

A MELL proof structure $(\mathcal{S}, B, \lambda)$ with boxes b_1, \dots, b_n is *DR-correct* if, for all $i \in \{0, \dots, n\}$, and for all switching S of $G_{(\mathcal{S}, B, \lambda)}^i$, the graph $S(G_{(\mathcal{S}, B, \lambda)}^i)$ is acyclic and connected.

A MELL proof structure $(\mathcal{S}, B, \lambda)$ with boxes b_1, \dots, b_n is *contractile* if $\forall i \in \{0, \dots, n\}$, $G_{(\mathcal{S}, B, \lambda)}^i \rightarrow_c^*$

•.

Theorem 1.12. [GM01] A MELL proof structure $(\mathcal{S}, B, \lambda)$ is a MELL proof net iff $(\mathcal{S}, B, \lambda)$ is DR-correct iff $(\mathcal{S}, B, \lambda)$ is contractile. \square

We define the following decision problem MELL-CORR:

GIVEN: A MELL proof structure $(\mathcal{S}, B, \lambda)$

PROBLEM: Is $(\mathcal{S}, B, \lambda)$ a MELL proof net?

1.3 MALL

We recall (and adapt to our formalism) the notion of MALL proof structures and proof nets defined in [HvG05]. The definition of MALL formulae follows that of MLL formulae in Section 1.1, with the additive connectives \oplus and $\&$, duals under De Morgan laws:

$$\text{MALL:} \quad F ::= A \mid A^\perp \mid F \otimes F \mid F \wp F \mid F \oplus F \mid F \& F$$

The MALL sequent calculus contains the rules of the MLL sequent calculus, as well as the following rules:

$$\frac{\vdash \Gamma, A}{\vdash \Gamma, A \oplus B} \oplus_1 \quad \frac{\vdash \Gamma, B}{\vdash \Gamma, A \oplus B} \oplus_2 \quad \frac{\vdash \Gamma, A \quad \vdash \Gamma, B}{\vdash \Gamma, A \& B} \&$$

Definition 1.13. MALL *skeletons* are defined similarly to MLL skeletons (Definition 1.1), with the following additional nodes:

node label		\oplus		$\&$	
arity	edge labels	2	A, B	2	A, B
coarity	edge labels	1	$A \oplus B$	1	$A \& B$

Definition 1.14. Let \mathcal{S} be a MALL skeleton. An *additive resolution* of \mathcal{S} is any result of deleting one argument subtree of each additive (\oplus or $\&$) node in \mathcal{S} . A *$\&$ -resolution* of \mathcal{S} is any result of deleting one argument subtree of each $\&$ -node in \mathcal{S} .

A *linking* on a MALL skeleton \mathcal{S} is a set of disjoint links on \mathcal{S} such that its set of vertices is the set of leaves of an additive resolution of \mathcal{S} . Note that in the case where \mathcal{S} contains no additive node, this definition subsumes Definition 1.1. The additive resolution of \mathcal{S} induced by a linking λ is denoted $\mathcal{S} \downarrow \lambda$.

An additive resolution of \mathcal{S} naturally induces a MLL skeleton, and, for any linking λ , $(\mathcal{S} \downarrow \lambda, \lambda)$ induces a MLL proof structure. Denote by $G_{(\mathcal{S} \downarrow \lambda, \lambda)}$ the paired graph associated to it.

A MALL *proof structure* is (\mathcal{S}, Θ) , where \mathcal{S} is a MALL skeleton and Θ is a set of linkings on \mathcal{S} . The set of conclusions of a MALL proof structure is a MALL sequent.

Definition 1.15. A MALL *proof net* is a MALL proof structure inductively defined as follows:

- (ax) : $((\{A, A^\perp\}, \emptyset), \{\{(A, A^\perp)\}\})$ is a MALL proof net with conclusions A, A^\perp .
- \wp : if (\mathcal{S}, Θ) is a MALL proof net with conclusions Γ, A, B , then (\mathcal{S}', Θ) , where \mathcal{S}' is \mathcal{S} extended with a \wp -node of premises A and B , is a MALL proof-net with conclusions $\Gamma, A \wp B$.
- \otimes : if $(\mathcal{S}_1, \Theta_1)$ with conclusions Γ, A and $(\mathcal{S}_2, \Theta_2)$ with conclusions Δ, B are disjoint MALL proof nets, (\mathcal{S}, Θ) where \mathcal{S} is $\mathcal{S}_1 \uplus \mathcal{S}_2$ extended with a \otimes -link of premises A and B and Θ is $\{\lambda_1 \uplus \lambda_2, \lambda_1 \in \Theta_1, \lambda_2 \in \Theta_2\}$, is a MALL proof net with conclusions $\Gamma, A \otimes B, \Delta$.
- (cut) : if $(\mathcal{S}_1, \Theta_1)$ with conclusions Γ, A and $(\mathcal{S}_2, \Theta_2)$ with conclusions Δ, A^\perp are disjoint MALL proof nets, (\mathcal{S}, Θ) where \mathcal{S} is $\mathcal{S}_1 \uplus \mathcal{S}_2$ extended with a cut-link of premises A and A^\perp and Θ is $\{\lambda_1 \uplus \lambda_2, \lambda_1 \in \Theta_1, \lambda_2 \in \Theta_2\}$, is a MALL proof net with conclusions Γ, Δ .
- $\&$: if $(\mathcal{S} \uplus \mathcal{S}_A, \Theta_A)$, where \mathcal{S} (respectively \mathcal{S}_A) has conclusions Γ (resp. A) and $(\mathcal{S} \uplus \mathcal{S}_B, \Theta_B)$, where \mathcal{S}_B has conclusion B are MALL proof nets, then $(\mathcal{S} \uplus \mathcal{S}', \Theta_A \uplus \Theta_B)$, where \mathcal{S}' is $\mathcal{S}_A \uplus \mathcal{S}_B$ extended with a $\&$ -node of premises A and B , is a MALL proof net with conclusions $\Gamma, A \& B$.

- \oplus : for any MALL formula B , if (\mathcal{S}, Θ) is a MALL proof net with conclusions Γ, A , then (\mathcal{S}', Θ) , where \mathcal{S}' is \mathcal{S} extended with the syntactic tree of B and a \oplus node of premises A and B (respectively B and A), is a MALL proof net with conclusions $\Gamma, A \oplus B$ (resp. $\Gamma, B \oplus A$).

Again, MALL proof nets are induced by MALL sequent calculus proofs.

Definition 1.16. Let (\mathcal{S}, Θ) be a MALL proof structure.

Let W be a $\&$ -resolution of \mathcal{S} and let $\lambda \in \Theta$ be a linking on \mathcal{S} . We note $\lambda \sqsubseteq W$ if and only if every vertex of every link in λ is a leaf of W .

Let $\Lambda \subseteq \Theta$ be a set of linkings on \mathcal{S} .

Λ is said to *toggle* a $\&$ node $x_{\&}$ (respectively a \oplus node x_{\oplus}) of \mathcal{S} if there exists $\lambda_1, \lambda_2 \in \Lambda$ such that $x_{\&}^l \in \mathcal{S} \downarrow \lambda_1$ and $x_{\&}^r \in \mathcal{S} \downarrow \lambda_2$ (resp. $x_{\oplus}^l \in \mathcal{S} \downarrow \lambda_1$ and $x_{\oplus}^r \in \mathcal{S} \downarrow \lambda_2$).

Let $\mathcal{S} \downarrow \Lambda = \bigcup_{\lambda \in \Lambda} \mathcal{S} \downarrow \lambda$, and $G_{\mathcal{S} \downarrow \Lambda} = \bigcup_{\lambda \in \Lambda} G_{(\mathcal{S} \downarrow \lambda, \lambda)}$.

Let $x_{\&}$ be a $\&$ node in \mathcal{S} and a be an atom of \mathcal{S} . Let $\{\lambda_1, \lambda_2\} \subseteq \Lambda$. A *jump edge* $(x_{\&}, a)$ is admissible for $\{\lambda_1, \lambda_2\}$ if and only if

1. $x_{\&}$ is the *unique* $\&$ node toggled by $\{\lambda_1, \lambda_2\}$, and,
2. there exists a link $l = (a, b) \in \lambda_1 \setminus \lambda_2$.

Let $H_{\mathcal{S} \downarrow \Lambda}$ be $G_{\mathcal{S} \downarrow \Lambda}$ extended with all admissible jump edges for all $\{\lambda_1, \lambda_2\} \subseteq \Lambda$, and where $C(H_{\mathcal{S} \downarrow \Lambda})$ contains the premise - and jump - edges incident to all $\wp/\&$ nodes of $\mathcal{S} \downarrow \Lambda$. (the pair edges are actually tuples as in Remark 1.3) Let G be a paired graph. A *switching cycle* \mathcal{C} in G is a cycle in $S(G)$ for some switching S of G .

Theorem 1.17 (Correctness Criterion). [HvG05]

A MALL proof structure (\mathcal{S}, Θ) is a MALL proof net iff:

1. (RES): For every $\&$ -resolution W of \mathcal{S} , there exists a unique $\lambda \in \Theta$ such that $\lambda \sqsubseteq W$,
2. (MLL): For every $\lambda \in \Theta$, $(\mathcal{S} \downarrow \lambda, \lambda)$ is a MLL proof net, and
3. (TOG): For every $\Lambda \subseteq \Theta$ of two or more linkings, Λ toggles a $\&$ node $x_{\&}$ such that $x_{\&}$ does not belong to any switching cycle of $H_{\mathcal{S} \downarrow \Lambda}$. \square

We define the following decision problem MALL-CORR:

GIVEN: A MALL proof structure (\mathcal{S}, Θ)

PROBLEM: Is (\mathcal{S}, Θ) a MALL proof net?

1.4 Complexity Classes and Related Problems

Let us mention several major complexity classes below P , some of which having natural complete problems that we will use in this paper. Let us briefly recall some basic definitions and results.

Definition 1.18. Complexity classes:

- AC^0 (respectively AC^1) is the class of problems solvable by a uniform family of circuits of constant (resp. logarithmic) depth and polynomial size, with NOT gates and AND, OR gates of unbounded fan-in.
- L is the class of problems solvable by a deterministic Turing machine which only uses a logarithmic working space.
- NL (respectively $coNL$) is the class of problems solvable by a non-deterministic Turing machine which only uses a logarithmic working space, such that:
 1. If the answer is "yes," at least one (resp. all) computation path accepts.
 2. If the answer is "no," all (resp. at least one) computation paths reject.

Theorem 1.19. [Imm88, Sze87] $NL = coNL$. □

The following inclusion results are also well known:

$$AC^0 \subseteq L \subseteq NL \subseteq AC^1 \subseteq P,$$

where it remains unknown whether any of these inclusions is strict.

It is important to note that our NL -completeness results for MLL-CORR, MELL-CORR and MALL-CORR are under constant-depth (actually AC^0) reductions. From the inclusion above, it should be clear to the reader that the reductions lie indeed in a class small enough for being relevant. For a good exposition of constant-depth reducibility, see [CSV84].

In the sequel, we will often use the notion of a *path* in a directed -or undirected- graph. A path is a sequence of vertices such that there is an edge between any two consecutive vertices in the path. A path will be called *elementary* when any node occurs at most once in the path.

Let us now list some graph-theoretic problems that will be used in this paper.

IS TREE (IT): Given an undirected graph $G = (V, E)$, is it a tree?

IT is L -complete under constant-depth reductions [JLM97].

SOURCE-TARGET CONNECTIVITY (STCONN): Given a directed graph $G = (V, E)$ and two vertices s and t , is there a path from s to t in G ?

STCONN is NL -complete under constant-depth reductions [JLL76].

UNDIRECTED SOURCE-TARGET CONNECTIVITY (USTCONN): Given an undirected graph $G = (V, E)$ and two vertices s and t , do s and t belong to the same connected component of G ?

USTCONN is L -complete under constant-depth reductions [Rei05].

UNIVERSAL SOURCE DAG (SDAG): Given a directed graph $G = (V, E)$, is it acyclic and does there exist a source node s such that there is a path from s to each vertex?

Proposition 1.20. $SDAG \in NL$.

Proof. Given $G = (V, E)$ a directed graph, its acyclicity can be expressed as follows:

$$\forall (x, y) \in V^2 : \neg \text{STCONN}(G, x, y) \vee \neg \text{STCONN}(G, y, x).$$

Since $NL = coNL$ (Theorem 1.19) and $\text{STCONN} \in NL$, acyclicity is clearly in NL . Checking whether each vertex can be reached from a vertex s can also be done with STCONN subroutines, therefore SDAG is in NL . □

Proposition 1.21. *SDAG is $coNL$ -hard under constant-depth reductions.*

Proof. Let \mathcal{L} be any language in $coNL$. \mathcal{L} is then decided by a non-deterministic Turing machine M in space less than $k \log(n)$ on inputs of size n , for some $k \geq 0$.

As it is usual in complexity, we denote by a configuration of a single tape Turing machine the tuple (S, T, pos) , where S is the current state of the machine, T the content of its tape and pos the position of the scanning head on the tape. The size of a configuration is the size of the non-empty part of its tape.

Let \mathcal{C}_n be the set of configurations of M of size less or equal to $k \log(n)$, and define $T = |\mathcal{C}_n|$. Clearly, $T = \mathcal{O}(n^{2k})$ is an upper bound for the computation time of M on inputs of size n . Without loss of generality, we assume that every configuration of M has at least one outgoing transition, possibly towards itself, and that the result of the computation is given by the state reached by M after exactly T computation steps. A configuration is thus either accepting or rejecting.

Let us consider the following directed graph $G_n = (V_n, E_n)$, where:

- $V_n = \bigcup_{c \in \mathcal{C}_n, t \in [0, T]} \{(c, t)\} \cup \{c_A\} \cup \{c_R\} \cup \{s\}$.
- For $(c, t), (c', t+1) \in V_n$, $((c', t+1) \rightarrow (c, t)) \in E_n$ if and only if $c \rightarrow c'$ is a transition of M .
- For $c \in \mathcal{C}_n$, $(c_A \rightarrow (c, T)) \in E_n$ iff c is an accepting configuration of M .

- For $c \in \mathcal{C}_n$, $(c_R \rightarrow (c, T)) \in E_n$ iff c is a rejecting configuration of M .
- $(s \rightarrow c_A) \in E_n$, $(s \rightarrow c_R) \in E_n$.

A path $(c_1, t_1) \rightarrow \dots \rightarrow (c_k, t_k)$ in G_n follows by construction a sequence t_1, \dots, t_k that is strictly decreasing. Since there is no edge $(c, t) \rightarrow c_A$, $(c, t) \rightarrow c_R$ nor $(c, t) \rightarrow s$, it is then clear that G_n is acyclic.

Moreover, since every configuration of M has at least one outgoing transition, every vertex (c, t) , $t < T$ in G_n has at least one parent node $(c', t + 1)$. By induction on t , it follows that every vertex in G_n is reachable from s . Therefore G_n satisfies SDAG.

Let x be an input of size n to M . An initial configuration $c_x \in \mathcal{C}_n$ of M is naturally associated to this input x . Consider now the directed graph $H_n^x = G_n \cup \{(c_x, 0) \rightarrow c_R\}$.

Then, H_n^x satisfies SDAG if and only if $x \in \mathcal{L}$. Indeed, by Definition 1.18, $x \in \mathcal{L}$ if and only if there exists no computation path $c_x \rightarrow \dots \rightarrow c_r$ of length T in M , where c_r is a rejecting configuration. By construction of G_n , such a path corresponds to a path $(c_r, T) \rightarrow \dots \rightarrow (c_x, 0)$ in G_n . Then $x \in \mathcal{L}$ if and only if there exists no path $c_R \rightarrow \dots \rightarrow (c_x, 0)$ in G_n , if and only if H_n^x is acyclic. Since G_n satisfies SDAG, it follows that H_n^x satisfies SDAG if and only if $x \in \mathcal{L}$.

Moreover, it is well known that the configuration graph of a Turing machine can be computed with constant-depth circuits. Computing H_n^x from the configuration graph of M requires only purely local rewriting rules, that can all be performed in parallel. Therefore, H_n^x can also be computed with constant-depth circuits. \square

Propositions 1.20 and 1.21, and Theorem 1.19 yield the following result:

Theorem 1.22. *SDAG is NL-complete under constant-depth reductions.* \square

2 MLL and MELL

2.1 New Correctness Criteria for MLL and MELL

For a given paired graph, the following notion of dependency graph provides a partial order among its pair-nodes corresponding to some valid contraction sequences accordingly to rule R_1 of Figure 2. Lemmas 2.3 and 2.4 establish that a paired graph G is D-R correct if and only if the graph $G[\forall \mapsto \lambda \cdot]$ of Definition 1.2 is a tree and its dependency graph satisfies SDAG. This yields a new correctness criterion for MLL-CORR and MELL-CORR given by Theorem 2.6.

Definition 2.1. Let G be a paired graph. The *dependency graph* $D(G)$ of G is the directed graph (V_G, E_G) defined as follows:

- $V_G = \{v \mid v \text{ is a pair-node in } G\} \cup \{s\}$.
- Let x be a pair-node in G , with premise-nodes x_l and x_r . The edge $(s \rightarrow x)$ is in E_G if and only if:
 1. There exists an elementary path $p_x = x_l, \dots, x_r$ in $G[\forall \mapsto \lambda \cdot]$,
 2. $x \notin p_x$, and for all pair-node y in G , $y \notin p_x$.
- Let x be a pair-node in G , with premise-nodes x_l and x_r , and let $y \neq x$ be another pair-node in G . The edge $(y \rightarrow x)$ is in E_G if and only if:
 1. There exists an elementary path $p_x = x_l, \dots, x_r$ in $G[\forall \mapsto \lambda \cdot]$,
 2. $x \notin p_x$, and for every elementary path $p_x = x_l, \dots, x_r$ in $G[\forall \mapsto \lambda \cdot]$ with $x \notin p_x$, $y \in p_x$.

For examples of MLL proof structures, corresponding paired graphs and their dependency graphs, see Figure 3 below.

Lemma 2.2. *Let G and H be paired graphs, with $G \rightarrow_c H$. Then, $G[\forall \mapsto \lambda \cdot] \rightarrow_c^* H[\forall \mapsto \lambda \cdot]$, and $G[\forall \mapsto \lambda \cdot]$ is a tree if and only if $H[\forall \mapsto \lambda \cdot]$ is a tree.*

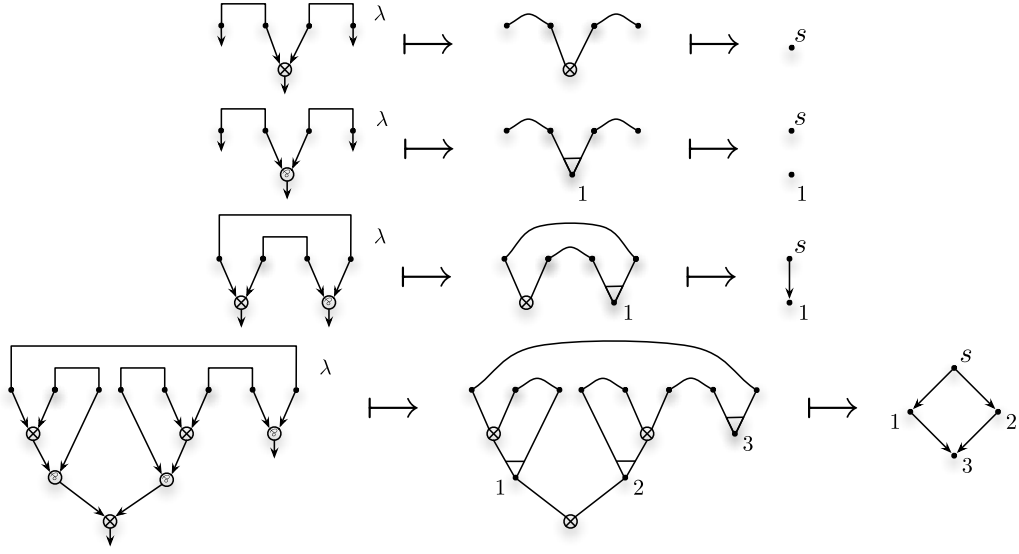


Figure 3: MLL proof structures, corresponding paired graph and dependency graphs, for the sequents $A^\perp, A \otimes B, B^\perp$ (correct), $A^\perp, A \wp B, B^\perp$ (incorrect), $A \otimes B, A^\perp \wp B^\perp$ (correct) and $((A \otimes B) \wp B^\perp) \otimes (C^\perp \wp (C \otimes D)), D^\perp \wp A^\perp$ (correct)

Proof. If $G \rightarrow_{R_1} H$ denote by v the redex pair-node in G , with premise w . The reduced pattern in H is the non-pair edge (v, w) , therefore $G[\forall \mapsto \lambda \cdot] = H[\forall \mapsto \lambda \cdot]$. If $G \rightarrow_{R_2} H$, it is clear that $G[\forall \mapsto \lambda \cdot] \rightarrow_{R_2} H[\forall \mapsto \lambda \cdot]$ with the same redex. It is also clear that rule \rightarrow_{R_2} preserves connectivity and acyclicity. \square

Lemma 2.3. *If $G \rightarrow_c^* \bullet$ then $D(G)$ satisfies SDAG.*

Proof. Since $\bullet[\forall \mapsto \lambda \cdot]$ is a tree, by Lemma 2.2 so is $G[\forall \mapsto \lambda \cdot]$. Therefore, for any pair-node x with premise-nodes x_l and x_r in G , there exists a unique elementary path $p_x = x_l - \dots - x_r$ in $G[\forall \mapsto \lambda \cdot]$. It follows by construction of $D(G)$ that x has at least one parent node in $D(G)$. Moreover, a path $x \rightarrow \dots \rightarrow y$ in $D(G)$ induces by construction an elementary path $x_l - \dots - y$ in $G[\forall \mapsto \lambda \cdot]$. Therefore a cycle $x \rightarrow \dots \rightarrow y, y \rightarrow \dots \rightarrow x$ in $D(G)$ induces a cycle $x_l - \dots - y, y_l - \dots - x$ in $G[\forall \mapsto \lambda \cdot]$. Since $G[\forall \mapsto \lambda \cdot]$ is a tree, $D(G)$ is acyclic. Since every vertex of $D(G)$ but s has at least one parent node and $D(G)$ is acyclic, $D(G)$ satisfies SDAG. \square

Lemma 2.4. *Let G be a paired graph such that $G[\forall \mapsto \lambda \cdot]$ is a tree. If the dependency graph $D(G)$ of G satisfies SDAG then $G \rightarrow_c^* \bullet$.*

Proof. let $d(v)$, the depth of a pair-node $v \in G$, be the length of the longest path from the source s of $D(G)$ to the vertex $v \in D(G)$. Assume that $D(G)$ satisfies SDAG, and let $X^d = \{x \text{ pair-node in } G \mid d(x) = d\}$ and $Y^d = \cup_{d' \leq d} X^{d'}$.

By induction on the depth we prove that there exists a sequence of contractions \mathcal{C}_d such that $G \rightarrow_{\mathcal{C}_d} G^d$ satisfies:

$$\text{Each pair-node } y \in G \text{ s.t. } d(y) \leq d \text{ is contracted in } G^d. \quad (1)$$

The proof by induction is as follows:

- For $d = 1$, let $x \in X^1$, with premise-nodes x_l and x_r . By definition of X^1 , there exists an elementary path $p_x = x_l - \dots - x_r$ in $G[\forall \mapsto \lambda \cdot]$ such that $x \notin p_x$ and for any pair-node y in $G[\forall \mapsto \lambda \cdot]$, $y \notin p_x$. The same holds for the path $p_x = x_l - \dots - x_r$ in G , with respect to any pair-node $y \in G$.

Let $\mathcal{E}_x^1 = \{e \text{ edge of } p_x \mid x \in X^1\}$. The set of contractions $\mathcal{R}_x^1 = \{e \rightarrow_c \bullet \mid e \in \mathcal{E}_x^1\}$ contracts the edges of p_x , and let $\mathcal{R}^1 = \cup_{x \in X^1} \mathcal{R}_x^1$. Clearly, $x_l = x_r \neq x$ in the contracted paired graph obtained from G by \mathcal{R}_x^1 . Since $x \notin p_y$ for any $y \in X^1$, the same holds for the contracted paired graph obtained from G by \mathcal{R}^1 .

Let \mathcal{C}_1 be the sequence \mathcal{R}^1 , followed by the set of contraction rules of the pair-nodes $x \in X^1$. Define G^1 such that $G \rightarrow^{\mathcal{C}_1} G^1$. It is clear that G^1 satisfies (1).

- Assume by induction that there exists a sequence of contractions \mathcal{C}_d such that $G \rightarrow^{\mathcal{C}_d} G^d$ satisfies (1).

Let $x \in X^{d+1}$, with premise-nodes x_l and x_r .

Since $G \rightarrow^{\mathcal{C}_d} G^d$ and $G[\forall \mapsto \lambda \cdot]$ is a tree, Lemma 2.2 applies:

$$G[\forall \mapsto \lambda \cdot] \rightarrow^{\mathcal{C}_d'} G^d[\forall \mapsto \lambda \cdot], \quad \text{and} \quad G^d[\forall \mapsto \lambda \cdot] \text{ is a tree.} \quad (2)$$

By definition of X^{d+1} , there exists an elementary path $p_x = x_l - \dots - x_r$ in $G[\forall \mapsto \lambda \cdot]$ such that $x \notin p_x$ and, for every pair-node $y \in G$ of depth $d(y) > d$, $y \notin p_x$.

Define p_x^d such that $p_x \rightarrow^{\mathcal{C}_d'} p_x^d$. By (2), p_x^d is an elementary path in $G^d[\forall \mapsto \lambda \cdot]$ such that $x \notin p_x^d$ and, for every pair-node $y \in G^d[\forall \mapsto \lambda \cdot]$ of depth $d(y) > d$, $y \notin p_x^d$. The same holds for p_x^d in G^d , with respect to any pair-node $y \in G^d$, since, by induction, for any pair-node $y \in G^d$, $d(y) > d$.

Let $\mathcal{E}_x^{d+1} = \{e \text{ edge of } p_x \mid x \in X^{d+1}\}$. The set of contractions $\mathcal{R}_x^{d+1} = \{e \rightarrow_c \bullet \mid e \in \mathcal{E}_x^{d+1}\}$ contracts the edges of p_x^d , and let $\mathcal{R}^{d+1} = \cup_{x \in X^{d+1}} \mathcal{R}_x^{d+1}$. Clearly, $x_l = x_r \neq x$ in the contracted paired graph obtained from G by \mathcal{R}_x^{d+1} . Since $x \notin p_y$ for any $y \in X^{d+1}$, the same holds for the contracted paired graph obtained from G by \mathcal{R}^{d+1} .

Let \mathcal{C}_{d+1} be the sequence \mathcal{C}_d , followed by \mathcal{R}_{d+1} , and followed by the set of contraction rules of the pair-nodes $x \in X^{d+1}$. Define G^{d+1} such that $G \rightarrow^{\mathcal{C}_{d+1}} G^{d+1}$. G^{d+1} satisfies (1).

Since $D(G)$ satisfies SDAG, the maximal depth $m = \max\{d(x) \mid x \in D(G)\}$ is well-defined and every pair-node x of G belongs to X^m . Therefore, $G \rightarrow^{\mathcal{C}_m} G^m$ and G^m satisfies (1). Since $G[\forall \mapsto \lambda \cdot]$ is a tree, by Lemma 2.2 so is $G^m[\forall \mapsto \lambda \cdot] = G^m$. It follows that $G \rightarrow_c^* \bullet$. \square

Define a paired-graph G to be *D-R-connected* if and only if, for any switching S of G , the switched graph $S(G)$ is connected. Lemmas 2.3 and 2.4 yields the following corollary:

Corollary 2.5. *A paired-graph G is D-R connected if and only if its dependency graph has a node s from which every node is reachable.*

Proof. An induction on the number of edges shows that G is D-R-connected if and only if there exists $G' \subseteq G$ D-R correct with the same set of vertices. By Lemmas 2.3 and 2.4, G' is D-R correct if and only if its dependency graph satisfies SDAG. Since the dependency graph of G' is a subgraph of the dependency graph of G , it follows that G is D-R connected if and only if its dependency graph has a node s from which every node is reachable. \square

Lemmas 2.3 and 2.4 and Theorems 1.5 and 1.12 imply the Theorem:

Theorem 2.6 (Correctness Criteria). *A MLL proof structure (\mathcal{S}, λ) is a MLL proof net if and only if:*

1. $D(G_{(\mathcal{S}, \lambda)})$ satisfies SDAG, and
2. $G_{(\mathcal{S}, \lambda)}[\forall \mapsto \lambda \cdot]$ is a tree.

A MELL proof structure $(\mathcal{S}, B, \lambda)$ with boxes b_1, \dots, b_n is a MELL proof net if and only if:

1. $\forall i \in \{0, \dots, n\}$, $D(G_{(\mathcal{S}, B, \lambda)}^i)$ satisfies SDAG, and
2. $\forall i \in \{0, \dots, n\}$, $G_{(\mathcal{S}, B, \lambda)}^i[\forall \mapsto \lambda \cdot]$ is a tree. \square

2.2 NL-Completeness of the Criteria for MLL and MELL

Proposition 2.7. *MLL-CORR is NL-hard under constant-depth reductions.*

Proof. We actually reduce SDAG to MLL-CORR. Let G be a directed graph, and consider the proof structure $(\mathcal{S}_G, \lambda_G)$ defined as follows (see Figure 4), and let $G_{(\mathcal{S}_G, \lambda_G)}$ be its associated paired graph:

1. To any vertex v of G , we associate a \otimes -node \bar{v} with parent nodes \bar{v}_{in} and \bar{v}_{out} .
2. If there are $i > 0$ in-going edges to v , \bar{v}_{in} is a \wp -node of arity i , with parent nodes $\bar{v}_{in}^1, \dots, \bar{v}_{in}^i$. $\bar{v}_{in}^1, \dots, \bar{v}_{in}^i$ are axiom-nodes. If v has no in-going edge, \bar{v}_{in} is an axiom-node, and λ_G contains a link $(\bar{v}_{in}, \bar{v}_{in}^2)$, where \bar{v}_{in}^2 is a conclusion of \mathcal{S}_G .
3. If there are $j > 0$ outgoing edges from v , \bar{v}_{out} is a \otimes -link of arity j , with parent links $\bar{v}_{out}^1, \dots, \bar{v}_{out}^j$. $\bar{v}_{out}^1, \dots, \bar{v}_{out}^j$ are axiom nodes. If v has no outgoing edge, \bar{v}_{out} is an axiom-node, and λ_G contains a link $(\bar{v}_{out}, \bar{v}_{out}^2)$, where \bar{v}_{out}^2 is a conclusion of \mathcal{S}_G .
4. Let $v \rightarrow w$ be an edge of G , and assume it is the k^{th} outgoing edge from v and the l^{th} in-going edge to w . To $v \rightarrow w$ we associate a link $(\bar{v}_{out}^k, \bar{w}_{in}^l)$ in λ_G .

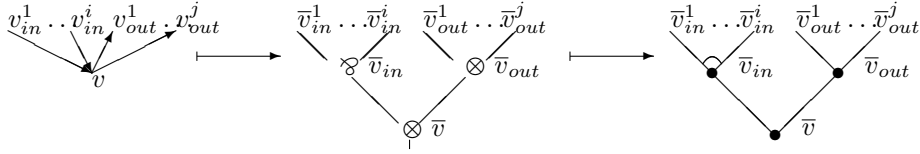


Figure 4: Construction of $(\mathcal{S}_G, \lambda_G)$ and $G_{(\mathcal{S}_G, \lambda_G)}$

It is quite clear that this reduction can be computed by constant-depth circuits. We now claim that $(\mathcal{S}_G, \lambda_G)$ is correct if and only if G satisfies SDAG.

Assume G contains a cycle. There exists then an elementary path $p = x_1 \rightarrow \dots \rightarrow x_l$, with $x_l \rightarrow x_1 \in G$. Then, for any edge $x_t \rightarrow x_{t+1} \in p$, there exists a switching of the pair-node $\bar{x}_{t+1}in$ in $G_{(\mathcal{S}_G, \lambda_G)}$, which connects \bar{x}_t and \bar{x}_{t+1} . Similarly for the edge $x_l \rightarrow x_1 \in G$. Since p is elementary, these pair-nodes are all different; therefore there exists cyclic switching of $G_{(\mathcal{S}_G, \lambda_G)}$ and $(\mathcal{S}_G, \lambda_G)$ is not correct.

It is clear that if G is acyclic, it has at least one node of arity 0. Moreover, if G is acyclic and has only one node of arity 0, a proof by induction shows that G satisfies SDAG.

Assume therefore that G is acyclic and has at least two nodes, r and s , of arity 0. Let S' be any switching of $G_{(\mathcal{S}_G, \lambda_G)}$, and assume that there exists an elementary path p from \bar{r} to \bar{s} in S' . Let $p' = \bar{r}, \bar{x}_1, \dots, \bar{x}_k, \bar{s}$ be the sequence of non pair-nodes of p corresponding to vertices of G . p' follows by construction edges of G , accordingly to their orientation or not. Since r and s have arity 0, there exist three nodes $\bar{x}_t, \bar{x}_{t+1}, \bar{x}_{t+2}$ in p' such that $(x_t \rightarrow x_{t+1})$ and $(x_{t+2} \rightarrow x_{t+1})$ are edges of G . By construction of $G_{(\mathcal{S}_G, \lambda_G)}$, \bar{x}_t and \bar{x}_{t+2} are then premise-nodes of the same pair-node $\bar{x}_{t+1}in$ in $G_{(\mathcal{S}_G, \lambda_G)}$, which contradicts that p is a path in S' . Therefore, S' is not connected, and $(\mathcal{S}_G, \lambda_G)$ is not correct.

Assume now that G satisfies SDAG and let $d(v)$, the depth of a vertex v of G , be the length of the longest path from the source s of G to v . Denote by G^d the subgraph of G consisting only in the vertices of depth less than d , and by $G_{(\mathcal{S}_G, \lambda_G)}^d$ the corresponding paired graph. It is easy to see that the rules of Figure 1 can be turned into a n-ary version, and that $G_{(\mathcal{S}_G, \lambda_G)}^{d+1}$ can be obtained from $G_{(\mathcal{S}_G, \lambda_G)}^d$ by these n-ary rules. By induction on d , it follows that $(\mathcal{S}_G, \lambda_G)$ is correct. \square

We denote by FL the class of functions computable in logarithmic working space (which is known to be stable under composition). Let $DEPGRAPH$ be the function: $G \mapsto D(G)$, which associates its dependency graph to a paired graph G .

Lemma 2.8. $\text{DEPGRAPH} \in FL$.

Proof. The following functions can easily be computed in FL :

- $G, x \in G \mapsto (G[\forall \mapsto \cdot]) \setminus \{x\}$
- $G, x \in G \mapsto (G[\forall \mapsto \cdot]) \setminus \{x\}$
- $G, x \in G, y \in G \mapsto (G[\forall \mapsto \cdot]) \setminus \{x, y\}$

Consider now the following algorithm for DEPGRAPH :

- ```

INPUT (G)
FOR ALL x pair-node in G , with premise-nodes x_l and x_r DO
 IF $\text{USTCONN}((G[\forall \mapsto \cdot]) \setminus \{x\}, x_l, x_r)$ THEN OUTPUT $(s \rightarrow x) \in D(G)$
FOR ALL (x pair-node in G , with premise-nodes x_l and x_r , y pair-node in G) DO
 IF $\neg \text{USTCONN}((G[\forall \mapsto \cdot]) \setminus \{x, y\}, x_l, x_r)$
 AND $\text{USTCONN}((G[\forall \mapsto \cdot]) \setminus \{x\}, x_l, x_r)$ THEN
 OUTPUT $(y \rightarrow \{x\}) \in D(G)$.

```
- $\text{USTCONN}((G[\forall \mapsto \cdot]) \setminus \{x\}, x_l, x_r)$  tests whether there exists an elementary path  $p_x = x_l - \dots - x_r$  such that  $x \notin p_x$  and, for all pair-node  $y$  in  $G$ ,  $y \notin p_x$ .
  - $\neg \text{USTCONN}((G[\forall \mapsto \cdot]) \setminus \{x, y\}, x_l, x_r)$  tests whether any elementary path  $p_x = x_l - \dots - x_r$  such that  $x \notin p_x$  contains  $y$ .
  - $\text{USTCONN}((G[\forall \mapsto \cdot]) \setminus \{x\}, x_l, x_r)$  tests whether there exists a path  $p_x = x_l - \dots - x_r$  in  $G'$  such that  $x \notin p_x$ . From the previous point, if such a path  $p_x$  exists,  $y \in p_x$ .

It follows that this algorithm computes  $\text{DEPGRAPH}$ . Since  $\text{USTCONN} \in L$  [Rei05], this algorithm belongs to  $FL^L$  (the class of functions computable in logspace with oracles in  $L$ ). Since  $FL^L = FL$ ,  $\text{DEPGRAPH} \in FL$ .  $\square$

**Proposition 2.9.**  $\text{MELL} - \text{CORR} \in NL$ .

*Proof.* Let  $(\mathcal{S}, B, \lambda)$  be a  $\text{MELL}$ -proof structure with boxes  $b_1, \dots, b_n$ . Each function  $(\mathcal{S}, B, \lambda), i \in \{0, \dots, n\} \mapsto G_{(\mathcal{S}, B, \lambda)}^i$  can easily be computed in  $FL$ . Checking that  $G_{(\mathcal{S}, B, \lambda)}^i[\forall \mapsto \cdot]$  is a tree is doable in  $L$  since  $\text{IT} \in L$ . Checking that  $D(G_{(\mathcal{S}, B, \lambda)}^i)$  satisfies  $\text{SDAG}$  can be done in  $NL$ , by composing the function  $\text{DEPGRAPH}$  in  $FL$  (Lemma 2.8) with an  $NL$  algorithm for  $\text{SDAG}$  (Theorem 1.22).

Since the number of paired graphs  $G_{(\mathcal{S}, B, \lambda)}^i$  is linearly bounded, it suffices to sequentially perform these tasks for  $i = 0, \dots, n$ , with a counter  $i$  of logarithmic size.  $\square$

Note that the previous best algorithms for  $\text{MELL-CORR}$  [Laf95, Gue99] are not likely to be implemented in logarithmic space, since they require on-line modification of the structure they manipulate. The purpose of our criterion of Theorem 2.6 is precisely that it allows a space-efficient implementation, at the cost of non-linear (actually quadratic) time execution.

For  $\text{MLL-CORR}$ , the linear time algorithms for essential nets of [MO00, MO06] are actually  $NL$  algorithms. However, they do not yield  $NL$  algorithms for  $\text{MLL}$  proof structures, since the reduction they use is not computable in logarithmic space.

### 3 MALL

This section is devoted to the proof of the  $NL$ -completeness of  $\text{MALL-CORR}$ . The situation for  $\text{MALL}$  differs quite a lot from the situation for  $\text{MLL}$  and  $\text{MELL}$  in the sense that the size of a sequent and of a corresponding proof structure - or proof net - may be of different order. For  $\text{MLL}$  and  $\text{MELL}$ , it is clear that the size of a proof structure is linear in the size of its skeleton. Yet, for  $\text{MALL}$ , the situation is more complex: while some  $\text{MALL}$  proof structures and proof nets have size linear in the size of their skeleton (e.g. pure  $\text{MLL}$  proof structures), others have size exponential in the size of their skeleton. Define the following correct sequents:

$$\begin{aligned}
\Gamma_1 &= A_1^\perp \oplus \dots \oplus A_n^\perp, A_1 \& \dots \& A_n \\
\Gamma_2 &= A^\perp \oplus \dots \oplus A^\perp, A \& \dots \& A \\
\Sigma_1 &= A_1^\perp \otimes \dots \otimes A_n^\perp, A_1 \& A_1, \dots, A_n \& A_n \\
\Sigma_2 &= A^\perp \otimes \dots \otimes A^\perp, A \& A, \dots, A \& A.
\end{aligned}$$

For each of these sequents, the size of the corresponding cut-free skeleton is linear in  $n$ . The following table shows, for a cut-free MALL skeleton for each of these sequents, its number of additive resolutions,  $\&$ -resolutions and possible links. The last two lines show the number of links in any cut-free MALL proof net, and the number of different cut-free MALL proof nets for each of these sequents.

| sequent             | $\Gamma_1$ | $\Gamma_2$ | $\Sigma_1$ | $\Sigma_2$ |
|---------------------|------------|------------|------------|------------|
| # add-resolutions   | $n^2$      | $n^2$      | $2^n$      | $2^n$      |
| # $\&$ -resolutions | $n$        | $n$        | $2^n$      | $2^n$      |
| # links             | $n$        | $n^2$      | $2^n$      | $n!2^n$    |
| $ \Theta $          | $n$        | $n$        | $2^n$      | 2          |
| # $\Theta$          | 1          | $n^2$      | 1          | $n!$       |

This table illustrates how some very simple MALL sequents can yield very large MALL proof nets. These proof-nets are exemplified in Figures 5, 6 and 7 below. Here, the reader should keep in mind that the input to the MALL-CORR problem is actually a MALL proof structure, of size maybe much larger than the size of the corresponding sequent. Recall from Theorem 1.17 that a MALL proof structure is a positive input to MALL-CORR if and only if it satisfies Conditions (MLL), (RES) and (TOG). The  $NL$ -hardness of MALL-CORR follows directly from the  $NL$ -hardness of MLL-CORR (since MLL is a sub-system of MALL). The  $NL$ -membership of Condition (MLL) follows directly from the  $NL$ -membership of MELL-CORR (and thus of MELL-CORR). Therefore, proving the  $NL$ -membership of MALL-CORR requires to prove the  $NL$ -membership of (RES) and (TOG). We exhibit in this section algorithms for checking non-deterministically (RES) and (TOG) in space logarithmic in the size of the proof structure, which, in some cases, is actually polynomial in the size of the sequent.

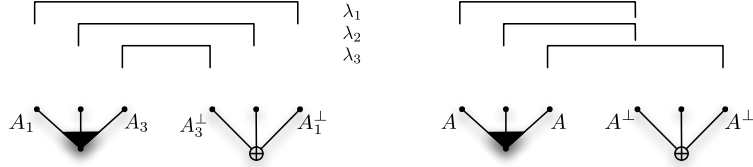


Figure 5: The MALL proof-net on  $\Gamma_1$ , and an example of proof-net on  $\Gamma_2$ , with  $n = 3$ .

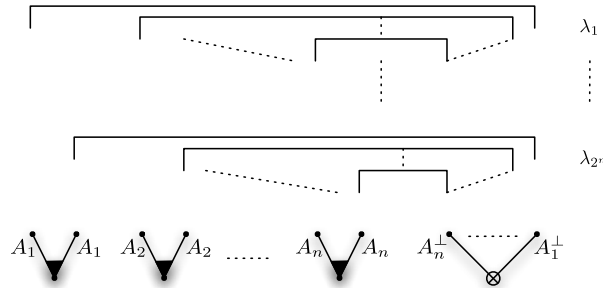


Figure 6: The MALL proof-net  $(\Sigma_1, \Theta_1)$  on  $\Sigma_1$ , with  $\Theta_1 = \bigcup_{i=1}^{2^n} \lambda_i$ .

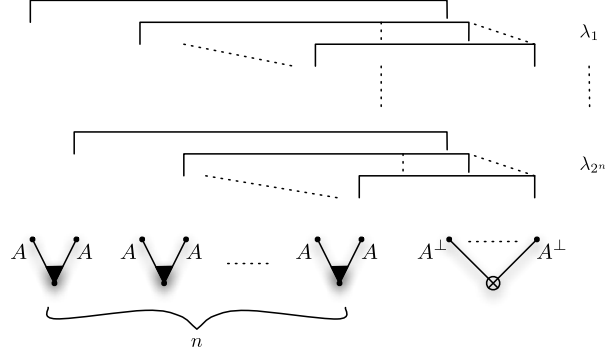


Figure 7: An example of MALL proof-net  $(\Sigma_2, \Theta_{n!})$  on  $\Sigma_2$ , with  $\Theta_{n!} = \bigcup_{i=1}^{2^n} \lambda_i$ . Note that the set  $\Theta_1$  of figure 6 yields another proof-net  $(\Sigma_2, \Theta_1)$  on  $\Sigma_2$ , as well as the  $n!$  possible combination of choices among the order in which the premises of the  $\otimes$  node are linked to the  $\&$  nodes.

### 3.1 checking (RES)

We recall Condition (RES) of Theorem 1.17: For every  $\&$ -resolution  $W$  of  $\mathcal{S}$ , there exists a unique  $\lambda \in \Theta$  such that  $\lambda \sqsubseteq W$ .

Let us illustrate the difficulty in checking (RES) on a simple example. Let us consider the proof-structure  $(\Sigma_1, \Theta)$ , where  $\Sigma_1$  is as above  $A_1^\perp \otimes \dots \otimes A_n^\perp, A_1 \& A_1, \dots, A_n \& A_n$ , and  $\Theta$  is a subset of  $\Theta_1$  of Figure 6 containing  $n^{\lceil \log(n) \rceil}$  linkings. The size of  $(\Sigma_1, \Theta)$  is therefore  $\mathcal{O}(n^{\lceil \log(n) \rceil})$ .

We have seen that the number of  $\&$ -resolutions of  $\Sigma_1$  is  $2^n$ . Enumerating (and explicitly describing) all  $\&$ -resolutions requires at least  $\Omega(n)$  space, and is not feasible in space  $\mathcal{O}(\log(n^{\lceil \log(n) \rceil})) = \mathcal{O}(\log(n)^2)$ . Therefore a *NL* algorithm for (RES) may not proceed by first plainly enumerating all  $\&$ -resolutions.

The idea of our algorithm is to define a notion of distance of edition on the  $\&$ -resolutions such that one can pass from any  $\&$ -resolution to any other  $\&$ -resolution with intermediate steps of distance at most one (Condition L1). Lemma 3.6 shows that (RES) fails if there exists a  $\&$ -resolution  $W$  with  $\lambda \sqsubseteq W$  at distance 1 to a  $\&$ -resolution  $W'$  with no  $\lambda' \sqsubseteq W'$  (Condition L3). Note however that, as on  $(\Sigma_1, \Theta)$ , the working space may not be large enough for describing explicitly the  $\&$ -resolutions: instead, a  $\&$ -resolution  $W$  with  $\lambda \sqsubseteq W$  is implicitly described by  $\lambda$ . The difficulty then is to describe a  $\&$ -resolution  $W'$  with no  $\lambda' \sqsubseteq W'$ . We establish in Lemma 3.9 that (RES) fails if there exists a  $\&$ -resolution  $W$  with  $\lambda \sqsubseteq W$  at distance 1 to a  $\&$ -resolution  $W'$  with no  $\lambda' \sqsubseteq W'$ , where moreover  $W'$  can be implicitly described by  $\lambda$  and some  $\&$ -node (Condition L4). Our algorithm enumerates (in logarithmic space) the  $\lambda$ 's and the  $\&$  nodes in search of such a configuration.

**Definition 3.1** (Condition L1). Let  $(\mathcal{S}, \Theta)$  be a MALL proof structure.

For any  $\&$ -resolution  $W$  of  $\mathcal{S}$ , let  $\text{switch}_W : \{x_\& : \& \text{ node of } \mathcal{S}\} \rightarrow \{l, r\}$  be the following function:

$$\text{switch}_W(x_\&) = \begin{cases} l & \text{if } x_\&^l \in W \text{ or } x_\& \notin W \\ r & \text{if } x_\&^r \in W. \end{cases}$$

Let  $\mathcal{W}_\mathcal{S}$  be the set of  $\&$ -resolutions of  $\mathcal{S}$ .

Let  $\mathcal{W}_\Theta = \{W \in \mathcal{W}_\mathcal{S} : \exists \lambda \in \Theta, \lambda \sqsubseteq W\}$ .

We define the following distance  $\text{Dist}$  on  $\mathcal{W}_\mathcal{S}$  by

$$\text{Dist}(W, W') = |\{x_\& \text{ \&-node of } \mathcal{S} : \text{switch}_W(x_\&) \neq \text{switch}_{W'}(x_\&)\}|.$$

Let  $\mathcal{W} \subseteq \mathcal{W}_\mathcal{S}$ . We say that  $\mathcal{W}$  satisfies Condition L1 if and only if:

$$\forall W_0, W_k \in \mathcal{W} \exists W_1, \dots, W_{k-1} \in \mathcal{W} \text{ s.t. } \text{Dist}(W_i, W_{i+1})_{0 \leq i < k \leq 1}.$$



**Lemma 3.2.**  $\mathcal{W}_S$  satisfies condition L1.

*Proof.* by induction on the skeleton  $\mathcal{S}$ . □

**Definition 3.3** (Condition L2). Let  $(\mathcal{S}, \Theta)$  be a MALL proof structure.

$(\mathcal{S}, \Theta)$  is said to satisfy Condition L2 if and only if  $\forall y_\oplus \oplus$  node in  $\mathcal{S}$ ,  $\forall \lambda_1, \lambda_2 \in \Theta$  that toggle  $y_\oplus$ , there exists a  $\&$  node  $x_\&$  also toggled by  $\{\lambda_1, \lambda_2\}$ .

**Lemma 3.4.** If  $(\mathcal{S}, \Theta)$  is a MALL proof net, then, it satisfies Condition L2.

*Proof:* By induction on  $(\mathcal{S}, \Theta)$ , along Definition 1.15. The only critical case is that of a  $\&$  rule: if  $(\mathcal{S} \uplus \mathcal{S}_A, \Theta_A)$ , where  $\mathcal{S}$  (respectively  $\mathcal{S}_A$ ) has conclusions  $\Gamma$  (resp.  $A$ ) and  $(\mathcal{S} \uplus \mathcal{S}_B, \Theta_B)$ , where  $\mathcal{S}_B$  has conclusion  $B$  are MALL proof nets, then  $(\mathcal{S} \uplus \mathcal{S}', \Theta_A \uplus \Theta_B)$ , where  $\mathcal{S}'$  is  $\mathcal{S}_A \uplus \mathcal{S}_B$  extended with a  $\&$ -node of premises  $A$  and  $B$ , is a MALL proof net with conclusions  $\Gamma, A \& B$ .

Two cases arise:

1. Assume there exist a  $\oplus$  node  $y_\oplus \in \mathcal{S}$ ,  $\lambda \in \Theta_A$ ,  $\lambda' \in \Theta_A$  such that  $\lambda, \lambda'$  toggle  $y_\oplus$ . Then the induction hypothesis on  $(\mathcal{S} \uplus \mathcal{S}_A, \Theta_A)$  ensures that there exists a  $\&$  node  $x_\& \in \mathcal{S} \uplus \mathcal{S}_A$  also toggled by  $\lambda, \lambda'$ . Similarly for  $\lambda \in \Theta_B$ ,  $\lambda' \in \Theta_B$ .
2. Assume there exist a  $\oplus$  node  $y_\oplus \in \mathcal{S}$ ,  $\lambda \in \Theta_A$ ,  $\lambda' \in \Theta_B$  such that  $\lambda, \lambda'$  toggle  $y_\oplus$ . Then the  $\&$  node of premises  $A$  and  $B$  in  $\mathcal{S}'$  is also toggled by  $\lambda, \lambda'$ . □

**Definition 3.5** (Condition L3). Let  $(\mathcal{S}, \Theta)$  be a MALL proof structure.

Let  $\lambda \in \Theta$ , and define  $\mathcal{S} \downarrow_\& \lambda = \{W \in \mathcal{W}_S : \lambda \sqsubseteq W\}$ .

Let  $x_\&$  be a  $\&$  node in  $\mathcal{S}$ .

$(\lambda, x_\&)$  are said to satisfy Condition L3 in  $(\mathcal{S}, \Theta)$  if and only if:

$$\begin{aligned} \exists W_+^\lambda \in \mathcal{S} \downarrow_\& \lambda, W_-^\lambda \in \mathcal{W}_S \setminus \mathcal{W}_\Theta \quad \text{s.t.} \quad \text{Dist}(W_+^\lambda, W_-^\lambda) = 1 \\ \text{and } \text{switch}_{W_+^\lambda}(x_\&) \neq \text{switch}_{W_-^\lambda}(x_\&). \end{aligned}$$

**Lemma 3.6.** Assume  $(\mathcal{S}, \Theta)$  is a MALL proof structure. Then,  $(\mathcal{S}, \Theta)$  satisfies (RES) of Theorem 1.17 if and only if:

1.  $\forall \lambda, \lambda' \in \Theta, \lambda \neq \lambda' \Rightarrow \mathcal{S} \downarrow \lambda \neq \mathcal{S} \downarrow \lambda'$ , and
2.  $\forall \lambda \in \Theta, \forall x_\& \&$  node in  $\mathcal{S}$ ,  $(\lambda, x_\&)$  does not satisfy L3 in  $(\mathcal{S}, \Theta)$ .

*Proof:*

1. Let  $W \in \mathcal{W}_\Theta$  and  $\lambda \in \Theta$  s.t.  $\lambda \sqsubseteq W$ . By induction on  $W$ , if there exists  $\lambda' \neq \lambda$  s.t.  $\lambda' \sqsubseteq W$ , then  $\mathcal{S} \downarrow \lambda = \mathcal{S} \downarrow \lambda'$ . It follows that (1) above is equivalent to the unicity, for any  $\&$ -resolution  $W$  of  $\mathcal{S}$ , of a  $\lambda \in \Theta$  such that  $\lambda \sqsubseteq W$ .
2. Assume that there exists a  $\&$ -resolution  $W$  of  $\mathcal{S}$  s.t.  $\forall \lambda \in \Theta, \lambda \not\sqsubseteq W$ . Then,  $\mathcal{W}_\Theta \subsetneq \mathcal{W}_S$ . Assume  $\Theta \neq \emptyset$ , then,  $\mathcal{W}_\Theta \neq \emptyset$ . Therefore there exists  $W_+ \in \mathcal{W}_\Theta$  and  $W_- \in \mathcal{W}_S \setminus \mathcal{W}_\Theta$ . By Lemma 3.4, there exists then  $W_1, \dots, W_k \in \mathcal{W}$  s.t.  $\text{Dist}(W_+, W_1) \leq 1$ ,  $\text{Dist}(W_i, W_{i+1})_{0 \leq i < k} \leq 1$ , and  $\text{Dist}(W_k, W_-) \leq 1$ . Since any of the  $W_i$  belongs either to  $\mathcal{W}_\Theta$  or to  $\mathcal{W}_S \setminus \mathcal{W}_\Theta$ , there exists  $W'_+, W'_- \in \{W_+, W_1, \dots, W_k, W_-\}$  such that  $\text{Dist}(W'_+, W'_-) = 1$ ,  $W'_+ \in \mathcal{W}_\Theta$  and  $W'_- \in \mathcal{W}_S \setminus \mathcal{W}_\Theta$ . Let  $\lambda \in \Theta$  such that  $\lambda \sqsubseteq W'_+$ , and  $x_\&$  be the  $\&$  node such that  $\text{switch}_{W'_+}(x_\&) \neq \text{switch}_{W'_-}(x_\&)$ . Clearly,  $(\lambda, x_\&)$  satisfy Condition L3.

Conversely, if there exists  $\lambda \in \Theta$  and  $x_\&$  a  $\&$  node in  $\mathcal{S}$  such that  $(\lambda, x_\&)$  satisfies L3 in  $(\mathcal{S}, \Theta)$ , then there exists a  $\&$ -resolution  $W$  of  $\mathcal{S}$  s.t.  $\forall \lambda \in \Theta, \lambda \not\sqsubseteq W$ . It follows that (2) above is equivalent to the existence, for any  $\&$ -resolution  $W$  of  $\mathcal{S}$ , of a  $\lambda \in \Theta$  such that  $\lambda \sqsubseteq W$ . □

**Definition 3.7** (Condition L4). Let  $(\mathcal{S}, \Theta)$  be a MALL proof structure. Let  $x_{\&}$  be a  $\&$  node in  $\mathcal{S}$ . Define:

$$\begin{aligned}\mathcal{W}_{x_{\&}}^l &= \{W \in \mathcal{W}_{\mathcal{S}} \text{ s.t. } \forall x'_{\&} \text{ s.t. there exists a path } x'_{\&} \rightarrow \dots \rightarrow x_{\&}^l, \text{switch}_W(x'_{\&}) = l\} \\ \mathcal{W}_{x_{\&}}^r &= \{W \in \mathcal{W}_{\mathcal{S}} \text{ s.t. } \forall x'_{\&} \text{ s.t. there exists a path } x'_{\&} \rightarrow \dots \rightarrow x_{\&}^r, \text{switch}_W(x'_{\&}) = l\}\end{aligned}$$

Let  $\lambda \in \Theta$ , and define

$$\begin{aligned}\text{Mirror}(\lambda, x_{\&}) &= \{W \in \mathcal{W}_{\mathcal{S}} \text{ s.t. } \exists W' \in \mathcal{S} \downarrow_{\&} \lambda \cap \mathcal{W}_{x_{\&}}^l \cap \mathcal{W}_{x_{\&}}^r : \\ &\quad \text{Dist}(W, W') = 1 \text{ and } \text{switch}_W(x_{\&}) \neq \text{switch}_{W'}(x_{\&})\}.\end{aligned}$$

$(\lambda, x_{\&})$  are said to satisfy Condition L4 in  $(\mathcal{S}, \Theta)$  if and only if:

$$\forall \lambda' \in \Theta, \forall W \in \text{Mirror}(\lambda, x_{\&}), \lambda' \not\sqsubseteq W.$$

**Lemma 3.8.** Assume  $(\mathcal{S}, \Theta)$  is a MALL proof structure satisfying Condition L2. Let  $\lambda \in \Theta$  and  $x_{\&}$  be a  $\&$  node in  $\mathcal{S}$  such that

1.  $(\lambda, x_{\&})$  satisfies Condition L3 in  $(\mathcal{S}, \Theta)$ , and
2.  $\forall y_{\oplus} \oplus$  node in  $\mathcal{S} \downarrow \lambda$ ,  $\forall \lambda' \in \Theta$  such that  $\lambda, \lambda'$  toggle  $y_{\oplus}$ ,  $x_{\&}$  is not toggled by  $\lambda, \lambda'$ .

Then,  $(\lambda, x_{\&})$  satisfies Condition L4 in  $(\mathcal{S}, \Theta)$ .

*Proof.* Let  $y_{\oplus}$  be a  $\oplus$  node in  $\mathcal{S} \downarrow \lambda$ . Without loss of generality, let assume that  $y_{\oplus}^l \in \mathcal{S} \downarrow \lambda$  and  $x_{\&}^l \in \mathcal{S} \downarrow \lambda$ . Assume  $(\lambda, x_{\&})$  satisfies Condition L3 in  $(\mathcal{S}, \Theta)$ :

$$\begin{aligned}\exists W_+^{\lambda} \in \mathcal{S} \downarrow_{\&} \lambda, W_-^{\lambda} \in \mathcal{W}_{\mathcal{S}} \setminus \mathcal{W}_{\Theta} \quad \text{s.t.} \quad & \text{Dist}(W_+^{\lambda}, W_-^{\lambda}) = 1 \\ & \text{and } \text{switch}_{W_+^{\lambda}}(x_{\&}) \neq \text{switch}_{W_-^{\lambda}}(x_{\&}).\end{aligned}$$

Let  $\theta_{\lambda} = \{\lambda_i \in \Theta : \lambda_i \sqsubseteq W_i \in \text{Mirror}(\lambda, x_{\&})\}$ .

Assume by contradiction that  $\theta_{\lambda} \neq \emptyset$ .

Let us show by contradiction that for all  $\lambda' \in \theta_{\lambda}$ ,  $y_{\oplus}^r \notin \mathcal{S} \downarrow \lambda'$ . Assume  $\exists \lambda' \in \theta_{\lambda}$ ,  $y_{\oplus}^r \in \mathcal{S} \downarrow \lambda'$ . Then  $\lambda, \lambda'$  toggle  $y_{\oplus}$ . By Condition L2, there exists a  $\&$  node  $x'_{\&} \neq x_{\&}$  also toggled by  $\lambda, \lambda'$ .

Assume without loss of generality that  $x_{\&}^l \in \mathcal{S} \downarrow \lambda$  and  $x_{\&}^r \in \mathcal{S} \downarrow \lambda'$ .

Since  $x_{\&}^l \in \mathcal{S} \downarrow \lambda$ , for all  $W \in \text{Mirror}(\lambda, x_{\&})$ ,  $\text{switch}_W(x_{\&}) = l$ . Since  $x_{\&}^r \in \mathcal{S} \downarrow \lambda'$ , for any  $W' \in \text{Mirror}(\lambda, x_{\&})$  s.t.  $\lambda' \sqsubseteq W'$ ,  $\text{switch}_{W'}(x_{\&}) = r$ : contradiction.

Therefore, for all  $\lambda' \in \theta_{\lambda}$ ,  $y_{\oplus}^r \notin \mathcal{S} \downarrow \lambda'$ .

Let  $\lambda' \in \theta_{\lambda}$ , and let  $x'_{\&}$  (respectively  $y'_{\oplus}$ ) be any  $\&$  node (resp.  $\oplus$  node) such that there exists no path  $x'_{\&} \rightarrow \dots \rightarrow x_{\&}$  (resp.  $y'_{\oplus} \rightarrow \dots \rightarrow y_{\oplus}$ ). Then, by induction on  $\mathcal{S}$ ,

$$\begin{aligned}x_{\&}^l \in \mathcal{S} \downarrow \lambda &\Rightarrow x_{\&}^l \in \mathcal{S} \downarrow \lambda', & y_{\oplus}^l \in \mathcal{S} \downarrow \lambda &\Rightarrow y_{\oplus}^l \in \mathcal{S} \downarrow \lambda', \\ x_{\&}^l \in \mathcal{S} \downarrow \lambda &\Rightarrow x_{\&}^l \in \mathcal{S} \downarrow \lambda', & y_{\oplus}^l \in \mathcal{S} \downarrow \lambda &\Rightarrow y_{\oplus}^l \in \mathcal{S} \downarrow \lambda', \\ x_{\&}^r \in \mathcal{S} \downarrow \lambda &\Rightarrow x_{\&}^r \in \mathcal{S} \downarrow \lambda', & y_{\oplus}^r \in \mathcal{S} \downarrow \lambda &\Rightarrow y_{\oplus}^r \in \mathcal{S} \downarrow \lambda'.\end{aligned}$$

It follows that  $\lambda' \sqsubseteq W_-^{\lambda}$ : contradiction.  $\square$

**Lemma 3.9.** Assume  $(\mathcal{S}, \Theta)$  is a MALL proof structure satisfying L2. Let  $\lambda \in \Theta$  and  $x_{\&}$  be a  $\&$  node in  $\mathcal{S}$  such that

1.  $(\lambda, x_{\&})$  satisfy Condition L3 in  $(\mathcal{S}, \Theta)$ , and
2.  $\exists y_{\oplus} \oplus$  node in  $\mathcal{S} \downarrow \lambda$ , and  $\lambda' \in \Theta$  such that  $\lambda, \lambda'$  toggle both  $y_{\oplus}$  and  $x_{\&}$ .

Then, there exists  $x'_{\&}$   $\&$  node in  $\mathcal{S}$  such that  $(\lambda', x'_{\&})$  satisfies Condition L4 in  $(\mathcal{S}, \Theta)$ .

Proof: By induction on the maximal number of  $\&$  and  $\oplus$  nodes traversed along a path  $x \rightarrow \dots \rightarrow x_{\&}$  or  $x \rightarrow \dots \rightarrow y_{\oplus}$  in  $\mathcal{S}$ . Since  $\mathcal{S}$  is acyclic, this number is well defined. Assume  $(\lambda, x_{\&})$  satisfies Condition L3 in  $(\mathcal{S}, \Theta)$ :

$$\begin{aligned} \exists W_+^\lambda \in \mathcal{S} \downarrow_{\&} \lambda, W_-^\lambda \in \mathcal{W}_{\mathcal{S}} \setminus \mathcal{W}_{\Theta} \quad \text{s.t.} \quad \text{Dist}(W_+^\lambda, W_-^\lambda) = 1 \\ \text{and } \text{switch}_{W_+^\lambda}(x_{\&}) \neq \text{switch}_{W_-^\lambda}(x_{\&}). \end{aligned}$$

Without loss of generality, assume  $y_{\oplus}^l \in \mathcal{S} \downarrow \lambda$  and  $x_{\&}^l \in \mathcal{S} \downarrow \lambda$ .

Let  $\theta_\lambda = \{\lambda_i \in \Theta : \lambda_i \sqsubseteq W_i \in \text{Mirror}(\lambda, x_{\&})\}$ . If there is no  $\&$  or  $\oplus$  node along any path  $x \rightarrow \dots \rightarrow x_{\&}$  or  $x \rightarrow \dots \rightarrow y_{\oplus}$ ,  $\theta_\lambda = \emptyset$ . If  $\theta_\lambda = \emptyset$ ,  $(\lambda, x_{\&})$  satisfies Condition L4 in  $(\mathcal{S}, \Theta)$ . Assume in the following that  $\theta_\lambda \neq \emptyset$ .

1. Let  $y_{\oplus}'$  be a  $\oplus$  node in  $\mathcal{S} \downarrow \lambda$  such that there exists no path  $y_{\oplus}' \rightarrow \dots \rightarrow y_{\oplus}$  and no path  $y_{\oplus}' \rightarrow \dots \rightarrow x_{\&}$ .

Let us show by contradiction that  $y_{\oplus}'$  is toggled by no  $(\lambda, \lambda_i)$ ,  $\lambda_i \in \theta_\lambda$ .

Assume  $y_{\oplus}'$  is toggled by  $(\lambda, \lambda_i)$ ,  $\lambda_i \in \theta_\lambda$ , and, without loss of generality,  $y_{\oplus}'^l \in \mathcal{S} \downarrow \lambda$ ,  $y_{\oplus}'^r \in \mathcal{S} \downarrow \lambda_i$ . Then, by Condition L2, there exists a  $\&$  node  $x_{\&}' \in \mathcal{S} \downarrow \lambda \cap \mathcal{S} \downarrow \lambda_i$  toggled by  $(\lambda, \lambda_i)$ , and, without loss of generality,  $x_{\&}'^l \in \mathcal{S} \downarrow \lambda$  and  $x_{\&}'^r \in \mathcal{S} \downarrow \lambda_i$ . Let  $W_i'$  be any  $\&$ -resolution such that  $\lambda_i \sqsubseteq W_i'$ :  $\forall W \in \mathcal{S} \downarrow_{\&} \lambda \cap \mathcal{W}_{x_{\&}}^l \cap \mathcal{W}_{x_{\&}}^r$ ,  $x_{\&}'^l \in W$ ,  $x_{\&}'^l \in W$ ,  $x_{\&}'^r \in W_i'$ ,  $x_{\&}'^r \in W_i'$ , and  $\text{Dist}(W, W') \geq 1$ . Therefore,  $W_i'$  cannot possibly be in  $\text{Mirror}(\lambda, x_{\&})$ , which contradicts the hypothesis that  $y_{\oplus}'$  is toggled by  $(\lambda, \lambda_i)$ ,  $\lambda_i \in \theta_\lambda$ .

2. By Condition L3,  $\forall \lambda_i \in \theta_\lambda$ ,  $\exists (x_i, y_i) \in \lambda_i : x_i \notin W_-^\lambda$ . Let us show that  $\forall (x_i, y_i) \in \lambda_i \in \theta_\lambda$ ,  $x_i \notin W_-^\lambda$ , there exists a path  $x_i \rightarrow \dots \rightarrow y_{\oplus}'$  or a path  $x_i \rightarrow \dots \rightarrow x_{\&}^r$ .

Assume there exists no such path. For any  $\oplus$  node  $y_{\oplus}'$  such that there exists a path  $x_i \rightarrow \dots \rightarrow y_{\oplus}'$ , there exists no path  $y_{\oplus}' \rightarrow \dots \rightarrow y_{\oplus}$  and no path  $y_{\oplus}' \rightarrow \dots \rightarrow x_{\&}$ . By (1) above,  $y_{\oplus}'$  is toggled by no  $(\lambda, \lambda_i)$ ,  $\lambda_i \in \theta_\lambda$ . Moreover, for any  $\&$  node  $x_{\&}'$  such that there exists a path  $x_i \rightarrow \dots \rightarrow x_{\&}'$ , there exists no path  $x_{\&}' \rightarrow \dots \rightarrow x_{\&}$ . By definition of  $\theta_\lambda$ ,  $x_{\&}'$  is then toggled by no  $(\lambda, \lambda_i)$ ,  $\lambda_i \in \theta_\lambda$ , and  $x_i \in \mathcal{S} \downarrow \lambda$ . Therefore,  $\forall W' \in \mathcal{S} \downarrow_{\&} \lambda$ ,  $x_i \in W'$ . By Condition L3, there exists  $W_+^\lambda \in \mathcal{S} \downarrow_{\&} \lambda$  s.t.  $\text{Dist}(W_+^\lambda, W_-^\lambda) = 1$  and  $\text{switch}_{W_+^\lambda}(x_{\&}) \neq \text{switch}_{W_-^\lambda}(x_{\&})$ . Since  $x_i \in W_+^\lambda$  and since there exists no path  $x_i \rightarrow \dots \rightarrow x_{\&}$ , it follows that  $x_i \in W_-^\lambda$ : contradiction.

3. By hypothesis,  $W_+^\lambda \in \mathcal{S} \downarrow_{\&} \lambda$ , and  $\text{switch}_{W_+^\lambda}(x_{\&}) = l$ . Since  $\text{Dist}(W_+^\lambda, W_-^\lambda) = 1$  and  $\text{switch}_{W_-^\lambda}(x_{\&}) = r$ , it follows that  $W_+^\lambda \in \mathcal{W}_{x_{\&}}^l \cap \mathcal{W}_{x_{\&}}^r$ , and therefore  $W_-^\lambda \in \text{Mirror}(\lambda, x_{\&})$ .
4. It is clear that  $\mathcal{S} \downarrow_{\&} \lambda$ ,  $\mathcal{W}_{x_{\&}}^l$  and  $\mathcal{W}_{x_{\&}}^r$  satisfy condition L1. Therefore, so does  $\text{Mirror}(\lambda, x_{\&})$ . Since  $W_-^\lambda \in \text{Mirror}(\lambda, x_{\&})$  and  $\theta_\lambda \neq \emptyset$ , there exist  $W_+^{\lambda_i}, W_-^{\lambda_i} \in \text{Mirror}(\lambda, x_{\&})$ ,  $\lambda_i \in \theta_\lambda$  such that  $\lambda_i \sqsubseteq W_+^{\lambda_i}, W_-^{\lambda_i} \in \mathcal{W}_{\mathcal{S}} \setminus \mathcal{W}_{\Theta}$  and  $\text{Dist}(W_+^{\lambda_i}, W_-^{\lambda_i}) = 1$ . Let  $x_{\&}'$  be the unique  $\&$  node in  $\mathcal{S}$  such that  $\text{switch}_{W_+^{\lambda_i}}(x_{\&}') \neq \text{switch}_{W_-^{\lambda_i}}(x_{\&}')$ . By (2) above, there exists a path  $x_{\&}' \rightarrow \dots \rightarrow y_{\oplus}$ . If there exists a  $\oplus$  node  $y_{\oplus}'$  in  $\mathcal{S} \downarrow \lambda_i$  and  $\lambda_j \in \theta_\lambda$  such that  $\lambda_i, \lambda_j$  toggle both  $x_{\&}'$  and  $y_{\oplus}'$ , by (1) above, there exists a path  $y_{\oplus}' \rightarrow \dots \rightarrow y_{\oplus}$  or a path  $y_{\oplus}' \rightarrow \dots \rightarrow x_{\&}$ . Therefore we can apply the induction hypothesis to conclude that  $(\lambda', x_{\&}')$  satisfies Condition L4 in  $(\mathcal{S}, \Theta)$ .  $\square$

**Proposition 3.10.** *Assume  $(\mathcal{S}, \Theta)$  is a MALL proof structure. Then,  $(\mathcal{S}, \Theta)$  satisfies (RES) of Theorem 1.17 if and only if:*

1.  $\forall \lambda, \lambda' \in \Theta$ ,  $\lambda \neq \lambda' \Leftrightarrow \mathcal{S} \downarrow \lambda \neq \mathcal{S} \downarrow \lambda'$ ,
2.  $(\mathcal{S}, \Theta)$  satisfies Condition L2, and
3.  $\forall \lambda \in \Theta$ ,  $\forall x_{\&} \&$  node in  $\mathcal{S}$ ,  $(\lambda, x_{\&})$  does not satisfy L4 in  $(\mathcal{S}, \Theta)$ .

*Proof.* Apply Lemmas 3.6, 3.8 and 3.9.  $\square$

A consequence of proposition 3.10 is a *NL* algorithm deciding whether a given MALL proof structure satisfies (RES). Indeed, by Proposition 2.9, (1) can be checked in *NL*, and Conditions L2 and L4 can easily be checked in *NL* by parsing the set of linkings and the skeleton.

### 3.1.1 Checking (TOG)

We recall Condition (TOG) of Theorem 1.17:

For every  $\Lambda \subseteq \Theta$  of two or more linkings,  $\Lambda$  toggles a  $\&$  node  $x_{\&}$  such that  $x_{\&}$  does not belong to any switching cycle of  $H_{S|\Lambda}$ .

Checking Condition (TOG) in non-deterministic logarithmic space involves two difficulties, which we address in this section:

1. The number of sets  $\Lambda \subseteq \Theta$  of two or more linkings is exponential in the size of  $\Theta$ , i.e. exponential in the size of the input in the worst case. Consider for instance the sequent  $\Gamma = A \& \dots \& A, A^\perp$  of figure 8 below: a proof-net  $(\Gamma, \Theta)$  contains  $n$  linkings, each linking containing a single link. The number of sets  $\Lambda \subseteq \Theta$  of two or more linkings is then  $2^n - n - 1$ . Clearly, there is no possibility to enumerate all the sets  $\Lambda \subseteq \Theta$  of two or more linkings in logarithmic space<sup>2</sup>. Lemma 3.12 below shows that it is actually enough to consider only a quadratic number of well chosen such sets of linkings.

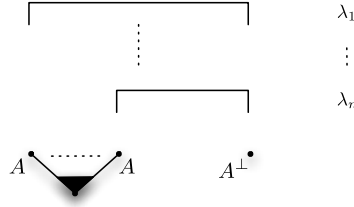


Figure 8: A proof-net  $(\Gamma, \Theta)$ , with  $\Theta = \bigcup_{i=1}^n \lambda_i$ .

2. given a set  $\Lambda \subseteq \Theta$  of two or more linkings and a  $\&$  node  $x_{\&}$  toggled by  $\Lambda$ , it remains to be checked whether  $x_{\&}$  belongs to a switching cycle of  $H_{S|\Lambda}$ . In the worst case, the number of switched graphs of  $H_{S|\Lambda}$  to be investigated may be also exponential in the size of the input. Moreover, it is unclear whether  $H_{S|\Lambda}$  enjoys properties such as D-R correctness that allow space-efficient algorithms. Lemma 3.17 below shows that the switching cycles of  $H_{S|\Lambda}$  are actually the switching cycles of a graph  $I_{S|\Lambda}$  which, in turns, enjoys the property of being D-R connected.

The two points above are necessary step-stones towards an *NL* algorithm for condition (TOG) exhibited in Proposition 3.18.

**Definition 3.11.** Let  $\{\lambda_1, \lambda_2\} \subseteq \Theta$ , we define  $\Theta_{\lambda_1, \lambda_2} = \{\lambda \in \Theta : \mathcal{S} \downarrow \lambda_1 \cap \mathcal{S} \downarrow \lambda_2 \subseteq \mathcal{S} \downarrow \lambda\}$ .

**Lemma 3.12.** Let  $(\mathcal{S}, \Theta)$  be a MALL proof structure satisfying (RES).

$(\mathcal{S}, \Theta)$  satisfies (TOG) if and only if, for all  $\{\lambda_1, \lambda_2\} \subseteq \Theta$ , there exists a  $\&$  node  $x_{\&}$  toggled by  $\lambda_1, \lambda_2$  such that  $x_{\&}$  does not belong to any switching cycle of  $H_{S|\Theta_{\lambda_1, \lambda_2}}$ .

*Proof.* In a first step, we show by induction on  $\mathcal{S} \setminus (\mathcal{S} \downarrow \lambda_1 \cap \mathcal{S} \downarrow \lambda_2)$  that, for all  $\Lambda \subseteq \Theta_{\lambda_1, \lambda_2}$  with at least two linkings,  $\Lambda$  toggles a  $\&$  node  $x'_{\&}$  such that  $x'_{\&}$  does not belong to any switching cycle of  $H_{S|\Lambda}$ .

Let  $\lambda_1, \lambda_2 \in \Theta$ ,  $x_{\&}$  a  $\&$  node toggled by  $\{\lambda_1, \lambda_2\}$  and  $\Lambda \subseteq \Theta_{\lambda_1, \lambda_2}$ . Then,  $H_{S|\Lambda} \subseteq H_{S|\Theta_{\lambda_1, \lambda_2}}$ , and the switching cycles of  $H_{S|\Lambda}$  are switching cycles of  $H_{S|\Theta_{\lambda_1, \lambda_2}}$ .

1. If  $\Lambda$  toggles  $x_{\&}$ , then  $x_{\&}$  belongs to no switching cycle of  $H_{S|\Lambda}$  (otherwise it would belong to a switching cycle of  $H_{S|\Theta_{\lambda_1, \lambda_2}}$ )

<sup>2</sup>It is mentioned in [HvG05] that it suffices to check (TOG) merely for *saturated* sets  $\Lambda$  of linkings only, namely, such that any strictly larger subset of  $\Theta$  toggles more  $\&$  nodes than  $\Lambda$ . Note however that the saturated sets of linkings are also exponentially many, and cannot be enumerated in logspace.

2. Assume  $\Lambda$  does not toggle  $x_{\&}$ . Then,  $(\mathcal{S} \downarrow \lambda_1 \cap \mathcal{S} \downarrow \lambda_2) \subsetneq \bigcap_{\lambda \in \Lambda} \mathcal{S} \downarrow \lambda$ .  
Let  $W_{\Lambda}^l$  be the  $\&$ -resolution of  $\mathcal{S}$  defined as follows:

$$\bigcap_{\lambda \in \Lambda} \mathcal{S} \downarrow \lambda \subseteq W_{\Lambda}^l, \text{ and}$$

$$\forall \& \text{ node } x'_{\&} \in \mathcal{S}, x'_{\&} \notin \bigcap_{\lambda \in \Lambda} \mathcal{S} \downarrow \lambda \Rightarrow x''_{\&} \text{ is erased in } W_{\Lambda}^l,$$

and  $W_{\Lambda}^r$  as follows:

$$\bigcap_{\lambda \in \Lambda} \mathcal{S} \downarrow \lambda \subseteq W_{\Lambda}^r, \text{ and}$$

$$\forall \& \text{ node } x'_{\&} \in \mathcal{S}, x'_{\&} \notin \bigcap_{\lambda \in \Lambda} \mathcal{S} \downarrow \lambda \Rightarrow x'^l_{\&} \text{ is erased in } W_{\Lambda}^r.$$

By Condition (RES), there exist  $\lambda^l, \lambda^r \in \Theta$  s.t.  $\lambda^l \sqsubseteq W_{\Lambda}^l$  and  $\lambda^r \sqsubseteq W_{\Lambda}^r$ . Then, clearly,  $\Lambda \subseteq \Theta_{\lambda^l, \lambda^r} \subsetneq \Theta_{\lambda_1, \lambda_2}$ . Since  $|\Theta_{\lambda^l, \lambda^r}| > 2$ , by Condition (RES),  $\Theta_{\lambda^l, \lambda^r}$  toggles a  $\&$  node  $x'_{\&} \neq x_{\&}$ . By construction,  $x'_{\&}$  is also toggled by  $\Lambda$ . The induction hypothesis on  $\Theta_{\lambda^l, \lambda^r}$ , and the arguments of (1) above yield that  $x'_{\&}$  belongs to no switching cycle of  $H_{\mathcal{S} \downarrow \Lambda}$ .

The second step is to show that there exist  $\lambda_1, \lambda_2 \in \Theta$  s.t.  $\Theta = \Theta_{\lambda_1, \lambda_2}$ . Consider  $W_l$  the  $\&$ -resolution of  $\mathcal{S}$  where all right premises of  $\&$  nodes are erased, and  $W_r$  the one where all left premises of  $\&$  nodes are erased. By Condition (RES), there exists  $\lambda_1, \lambda_2 \in \Theta$  such that  $\lambda_1 \sqsubseteq W_l$  and  $\lambda_2 \sqsubseteq W_r$ . It is clear that, for all  $\lambda \in \Theta$ ,  $\mathcal{S} \downarrow \lambda_1 \cap \mathcal{S} \downarrow \lambda_2 \subseteq \mathcal{S} \downarrow \lambda$ . Therefore,  $\Theta \subseteq \Theta_{\lambda_1, \lambda_2}$ .  $\square$

**Definition 3.13.** Let  $(\mathcal{S}, \Theta)$  be a MALL proof structure.

Let  $x_{\&}$  be a  $\&$  node in  $\mathcal{S}$ .  $x_{\&}$  is said to be *environment-free* if, for all  $\lambda \in \Theta$ , for all link  $(a, b) \in \lambda$ , there exists a path  $a \rightarrow \dots \rightarrow x_{\&}$  if and only if there exists a path  $b \rightarrow \dots \rightarrow x_{\&}$ . If  $x_{\&}$  is not environment-free, it is said to be *environment linked*.

**Lemma 3.14.** *If  $(\mathcal{S}, \Theta)$  is a MALL proof net then, for all  $\&$  node  $x_{\&}$ ,  $x_{\&}$  is environment-free if and only if, for any sequentialization of  $(\mathcal{S}, \Theta)$ , any  $\&$ -rule applied on  $x_{\&}$  has an empty environment  $\Gamma$ .*

*Proof.* Straightforward proof by induction.  $\square$

**Definition 3.15.** Let  $(\mathcal{S}, \Theta)$  be a MALL proof structure.

Let  $I_{\mathcal{S} \downarrow \Lambda}$  be  $G_{\mathcal{S} \downarrow \Lambda}$  extended with all admissible jump edges for all  $\{\lambda_1, \lambda_2\} \subseteq \Lambda$  and where  $C(I_{\mathcal{S} \downarrow \Lambda})$  contains the premise - and jump - edges incident to all  $\wp$  nodes and environment-linked  $\&$  nodes of  $\mathcal{S} \downarrow \Lambda$ , and the jump edges only incident to all environment-free  $\&$  nodes of  $\mathcal{S} \downarrow \Lambda$ .

**Lemma 3.16.** *If  $(\mathcal{S}, \Theta)$  is a MALL proof net then, for all  $\{\lambda_1, \lambda_2\} \subseteq \Theta$ ,  $I_{\mathcal{S} \downarrow \Theta_{\lambda_1, \lambda_2}}$  is D-R-connected.*

*Proof.* We actually prove the lemma for the graph  $I_{\mathcal{S} \downarrow \Theta_{\lambda_1, \lambda_2}}$  without jumps. An easy graph-theoretic proof by induction shows that adding the jumps does not D-R-Disconnect the paired graph.

The proof is by induction on  $(\mathcal{S}, \Theta)$ , along Definition 1.15. The only critical case is that of a  $\&$  rule on  $\Gamma, A \& B$ , where the  $\&$  node  $x_{\&}$  introduced by the rule is environment-linked and is toggled by  $\lambda_1, \lambda_2$ . Assume without loss of generality that  $x^l_{\&} \in \mathcal{S} \downarrow \lambda_1$  and  $x^r_{\&} \in \mathcal{S} \downarrow \lambda_2$ .

By Definition 1.15,  $\Theta = \Theta_A \uplus \Theta_B$ , and  $\mathcal{S}$  is  $\mathcal{S}_{\Gamma} \uplus \mathcal{S}_A \uplus \mathcal{S}_B$  (with respective conclusions  $\Gamma, A$  and  $B$ ) extended with  $x_{\&}$ , and  $(\mathcal{S}_{\Gamma} \uplus \mathcal{S}_A, \Theta_A)$ ,  $(\mathcal{S}_{\Gamma} \uplus \mathcal{S}_B, \Theta_B)$  are both MALL proof nets, and by Lemma 3.14,  $\mathcal{S}_{\Gamma} \neq \emptyset$ .

Let  $\Lambda_A = \{\lambda \in \Theta_A : \mathcal{S}_{\Gamma} \downarrow \lambda_1 \cap \mathcal{S}_{\Gamma} \downarrow \lambda_2 \subseteq \mathcal{S}_{\Gamma} \downarrow \lambda\}$  and  $\Lambda_B = \{\lambda \in \Theta_B : \mathcal{S}_{\Gamma} \downarrow \lambda_1 \cap \mathcal{S}_{\Gamma} \downarrow \lambda_2 \subseteq \mathcal{S}_{\Gamma} \downarrow \lambda\}$ . Then, clearly,  $\Theta_{\lambda_1, \lambda_2} = \Lambda_A \uplus \Lambda_B$ ,  $\lambda_1 \in \Lambda_A$  and  $\lambda_2 \in \Lambda_B$ .

Let  $W_1^l$  be the  $\&$ -resolution of  $\mathcal{S}$  defined as follows:

$$\begin{aligned} \mathcal{S} \downarrow \lambda_1 \cap \mathcal{S} \downarrow \lambda_2 &\subseteq W_1^l, \\ \forall \& \text{ node } x'_{\&} \in \mathcal{S}, x'_{\&} \notin \mathcal{S} \downarrow \lambda_1 \cap \mathcal{S} \downarrow \lambda_2 &\Rightarrow x'^l_{\&} \text{ is erased in } W_1^l, \text{ and} \\ &x^r_{\&} \text{ is erased in } W_1^l, \end{aligned}$$

and  $W_1^r$  as follows:

$$\begin{aligned} \mathcal{S} \downarrow \lambda_1 \cap \mathcal{S} \downarrow \lambda_2 &\subseteq W_1^r, \\ \forall \& \text{ node } x'_{\&} \in \mathcal{S}, x'_{\&} \notin \mathcal{S} \downarrow \lambda_1 \cap \mathcal{S} \downarrow \lambda_2 &\Rightarrow x'^r_{\&} \text{ is erased in } W_1^r, \text{ and} \\ &x^l_{\&} \text{ is erased in } W_1^r. \end{aligned}$$

Then, by Condition (RES), there exists  $\lambda_1^l, \lambda_1^r \in \Theta$  s.t.  $\lambda_1^l \subseteq W_1^l$  and  $\lambda_1^r \subseteq W_1^r$ . Moreover,  $\lambda_1^l \in \Theta_A$ ,  $\lambda_1^r \in \Theta_A$  and  $\mathcal{S} \downarrow \lambda_1^l \cap \mathcal{S} \downarrow \lambda_1^r = \mathcal{S} \downarrow \lambda_1 \cap \mathcal{S} \downarrow \lambda_2$ . Therefore,  $\Lambda_A = \Theta_{\lambda_1^l, \lambda_1^r}$ .

Similarly, there exists  $\lambda_2^l, \lambda_2^r \in \Theta$  s.t.  $\Lambda_B = \Theta_{\lambda_2^l, \lambda_2^r}$ .

By induction hypothesis,  $I_{\mathcal{S} \downarrow \Theta_{\lambda_1, \lambda_2}} = I_{\mathcal{S} \downarrow \Theta_{\lambda_1^l, \lambda_1^r}} \cup I_{\mathcal{S} \downarrow \Theta_{\lambda_2^l, \lambda_2^r}}$  where  $I_{\mathcal{S} \downarrow \Theta_{\lambda_1^l, \lambda_1^r}}$  and  $I_{\mathcal{S} \downarrow \Theta_{\lambda_2^l, \lambda_2^r}}$  are both D-R-connected.

Moreover, by Condition (RES), neither  $I_{\mathcal{S} \downarrow \Theta_{\lambda_1^l, \lambda_1^r}}$  nor  $I_{\mathcal{S} \downarrow \Theta_{\lambda_2^l, \lambda_2^r}}$  contains a unary couple of edges except for  $x_{\&}$ . Therefore, for any switching  $S$  of  $I_{\mathcal{S} \downarrow \Theta_{\lambda_1, \lambda_2}}$ ,  $x_{\&}^l$  is connected through  $S(I_{\mathcal{S} \downarrow \Theta_{\lambda_1^l, \lambda_1^r}})$  to some vertex  $y \in I_{\mathcal{S} \downarrow \Theta_{\lambda_1^l, \lambda_1^r}} \cap I_{\mathcal{S} \downarrow \Theta_{\lambda_2^l, \lambda_2^r}} \neq \emptyset$ , and back to  $x_{\&}^r$  through  $S(I_{\mathcal{S} \downarrow \Theta_{\lambda_2^l, \lambda_2^r}})$ .  $\square$

**Lemma 3.17.** *Let  $(\mathcal{S}, \Theta)$  be a MALL proof structure satisfying (RES) and let  $\Lambda \subseteq \Theta$  with at least two linkings.*

*$\Lambda$  toggle a  $\&$  node  $x_{\&}$  such that  $x_{\&}$  belongs to a switching cycle of  $I_{\mathcal{S} \downarrow \Lambda}$  if and only if it belongs to a switching cycle of  $H_{\mathcal{S} \downarrow \Lambda}$ .*

*Proof.* Condition (RES) implies that no premise edge of any environment-free  $\&$  node belongs to any switching cycle of  $H_{\mathcal{S} \downarrow \Lambda}$ . Therefore, the switching cycles of  $H_{\mathcal{S} \downarrow \Lambda}$  are switching cycles of  $I_{\mathcal{S} \downarrow \Lambda}$ , hence the “if” direction. The “only if” direction proceeds from the fact that the switching cycles of  $I_{\mathcal{S} \downarrow \Lambda}$  are switching cycles of  $H_{\mathcal{S} \downarrow \Lambda}$ .  $\square$

Lemmas 3.12 and 3.17 yield the following proposition:

**Proposition 3.18.** *Let  $(\mathcal{S}, \Theta)$  be a MALL proof structure satisfying (RES).  $(\mathcal{S}, \Theta)$  satisfies (TOG) iff, for all  $\{\lambda_1, \lambda_2\} \subseteq \Theta$ ,  $\Theta_{\lambda_1, \lambda_2}$  toggles a  $\&$  node  $x_{\&}$  such that  $x_{\&}$  does not belong to any switching cycle of  $I_{\mathcal{S} \downarrow \Theta_{\lambda_1, \lambda_2}}$ .*  $\square$

**Proposition 3.19.** *Let  $(\mathcal{S}, \Theta)$  be a MALL proof structure satisfying (RES) and (MLL). The following algorithm decides whether  $(\mathcal{S}, \Theta)$  satisfies (TOG) in non-deterministic logarithmic space:*

```

FOR ALL $\lambda_1, \lambda_2 \in \Theta$
 COMPUTE $I_{\mathcal{S} \downarrow \Theta_{\lambda_1, \lambda_2}}$,
 COMPUTE $D(I_{\mathcal{S} \downarrow \Theta_{\lambda_1, \lambda_2}})$ the dependency graph of $I_{\mathcal{S} \downarrow \Theta_{\lambda_1, \lambda_2}}$,
 IF $\forall s \in D(I_{\mathcal{S} \downarrow \Theta_{\lambda_1, \lambda_2}}), \exists x \in D(I_{\mathcal{S} \downarrow \Theta_{\lambda_1, \lambda_2}})$ such that $\neg \text{STCONN}(s, x)$
 THEN REJECT
 ELSE
 LET $\text{tog} = \text{false}$
 FOR ALL $\&$ node $x_{\&}$ in \mathcal{S}
 LET $I_{x_{\&}}$ be $I_{\mathcal{S} \downarrow \Theta_{\lambda_1, \lambda_2}}[\forall \mapsto \cdot]$ without any premise -or jump- edge to $x_{\&}$,
 IF no premise-argument or jump-argument of $x_{\&}$ is connected to $x_{\&}$ in $I_{x_{\&}}$
 THEN $\text{tog} = \text{true}$
 END FOR ALL
 END IF
 IF $\text{tog} = \text{false}$ THEN REJECT
END FOR ALL
ACCEPT

```

*Proof.* By Proposition 3.19,  $(\mathcal{S}, \Theta)$  satisfies (TOG) if and only if, for all  $\{\lambda_1, \lambda_2\} \subseteq \Theta$ ,  $\Theta_{\lambda_1, \lambda_2}$  toggles a  $\&$  node  $x_{\&}$  such that  $x_{\&}$  does not belong to any switching cycle of  $I_{\mathcal{S}|\Theta_{\lambda_1, \lambda_2}}$ . By Lemma 3.16, if  $(\mathcal{S}, \Theta)$  satisfies (TOG), then  $I_{\mathcal{S}|\Theta_{\lambda_1, \lambda_2}}$  is D-R-connected, and, by Corollary 2.5, its dependency graph has a node  $s$  from which every node is reachable. Now, if  $I_{\mathcal{S}|\Theta_{\lambda_1, \lambda_2}}$  is D-R-connected, a  $\&$  node  $x_{\&}$  belongs to a switching cycle of  $I_{\mathcal{S}|\Theta_{\lambda_1, \lambda_2}}$  if and only if it belongs to a cycle of  $I_{\mathcal{S}|\Theta_{\lambda_1, \lambda_2}}[\forall \mapsto \lambda \cdot]$ , therefore the algorithm above decides whether  $(\mathcal{S}, \Theta)$  satisfies (TOG). It is clear that the enumeration of the  $\lambda_1, \lambda_2 \in \Theta$ , and the computation of  $I_{\mathcal{S}|\Theta_{\lambda_1, \lambda_2}}$  and  $D(I_{\mathcal{S}|\Theta_{\lambda_1, \lambda_2}})$  can be performed in logarithmic space. Since STCONN  $\in$  NL, the whole algorithm works in NL.  $\square$

Propositions 2.7, 2.9, 3.10 and 3.19 yield the following result:

**Theorem 3.20.** *MLL-CORR, MELL-CORR and MALL-CORR are NL-complete under constant-depth reductions.*  $\square$

## References

- [CSV84] Ashok K. Chandra, Larry J. Stockmeyer, and Uzi Vishkin. Constant depth reducibility. *SIAM J. Comput.*, 13(2):423–439, 1984.
- [Dan90] Vincent Danos. *Une application de la logique linéaire à l'étude des processus de normalisation (principalement de  $\lambda$ -calcul)*. PhD thesis, Université Denis Diderot, Paris 7, 1990.
- [dF03] Lorenzo Tortora di Falco. The additive multiboxes. *Annals of Pure and Applied Logic*, 120(1–3):65–102, 2003.
- [DR89] Vincent Danos and Laurent Regnier. The structure of multiplicatives. *Archive for Mathematical Logic*, 28(3):181–203, 1989.
- [Gir87] Jean-Yves Girard. Linear logic. *Theoretical Computer Science*, 50(1):1–102, 1987.
- [Gir96] Jean-Yves Girard. Proof-nets: The parallel syntax for proof-theory. In P. Agliano and A. Ursini, editors, *Logic and Algebra*, volume 180. Marcel Dekker, New York, 1996.
- [GM01] Stefano Guerrini and Andrea Masini. Parsing MELL proof nets. *Theoretical Computer Science*, 254(1–2):317–335, 2001.
- [Gue99] Stefano Guerrini. Correctness of multiplicative proof nets is linear. In *Proc. of the annual Symp. on Logic in Computer Science (LICS'99)*, pages 454–463. IEEE Computer Society Press, 1999.
- [HvG05] Dominic j. d. Hughes and Rob j. van Glabbeek. Proof nets for unit-free multiplicative-additive linear logic. *ACM Trans. Comput. Logic*, 6(4):784–842, 2005.
- [Imm88] Neil Immerman. Nondeterministic space is closed under complementation. *SIAM J. Comput.*, 17(5):935–938, 1988.
- [JLL76] Neil D. Jones, Y. Edmund Lien, and William T. Laaser. New problems complete for nondeterministic logspace. *Mathematical Syst. Theory*, 10:1–17, 1976.
- [JLM97] B. Jenner, K.-J. Lange, and P. McKenzie. Tree isomorphism and some other complete problems for deterministic logspace. DIRO 1059, Univ. de Montréal, 1997.
- [Kan92] Max I. Kanovich. Horn programming in linear logic is NP-complete. In *Proc. of the annual Symp. on Logic in Computer Science (LICS'92)*, pages 200–210. IEEE Computer Society Press, 1992.

- [Laf95] Yves Lafont. From proof-nets to interaction nets. In J.-Y. Girard, Y. Lafont, and L. Regnier, editors, *Advances in Linear Logic*, volume 222, pages 225–247. Cambridge University Press, 1995.
- [Mai06] Harry G. Mairson. Normalization bounds for multiplicative linear logic are axiom-sensitive. Presentation at GEOCAL’06 Workshop of Implicit Computational Complexity. Slides available at <http://www-lipn.univ-paris13.fr/~baillet/GEOCAL06/SLIDES/Mairson.pdf>, 2006.
- [Mel04] Paul-André Melliès. *A topological correctness criterion for non-commutative logic*, volume 316 of *London Mathematical Society Lecture Notes Series*. Cambridge University Press, 2004.
- [MO00] Andrzej Murawski and Luke Ong. Dominator trees and fast verification of proof nets. In *Proc. of the annual Symp. on Logic in Computer Science (LICS’00)*, pages 181–191. IEEE Computer Society Press, 2000.
- [MO06] Andrzej Murawski and Luke Ong. Fast verification of MLL proof nets via IMLL. *ACM Trans. Comput. Logic*, 7(3):473–498, 2006.
- [Rei05] Omer Reingold. Undirected st-connectivity in log-space. In Harold N. Gabow and Ronald Fagin, editors, *STOC*, pages 376–385. ACM, 2005.
- [Sze87] Róbert Szelepcsényi. The method of forcing for nondeterministic automata. *Bulletin of the EATCS*, 33:96–99, 1987.